HUMAN RESOURCE MANAGEMENT SYSTEM

(HRMS)

BONG MEE MEE

WEK 98041

Supervised by Puan Nor Edzan Haji Che Nasir

This project is submitted to the

Faculty of Computer Science and Information Technology

in fulfillment of the requirements for the

Degree of Bachelor of Computer Science

Session 2000/2001

TABLE OF CONTENT

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude to Puan Nor Edzan Haji Che Nasir, the supervisor of my final year project for being so kind, helpful and patient throughout the development of my project, Human Resource Management System.

I would also like to thank Puan Salimah Mokhtar, the moderator of this final year project for assisting me in providing brilliant ideas and opinions to build the system a successful one during viva section held previously.

A special thanks to Cik Iza Inora, Accounting and Administrative Executive of LK Solutions Asia Pacific for being kind to assist me in providing me valuable information about human resources management in the organization.

Finally, I would like to express my thanks to all my coursemates, especially Kenny Chong Yuen Beng who are willing to spend time to discuss about how to produce an ease of use, secure and effective system application.

ABSTRACT

HRMS (Human Resource Management System) is an interactive, web-based application developed using ASP (Active Server Pages). It is designed to meet the need of accessing human resource information electronically by the employees of an organization. This system can be reach through network, only by authorized users.

In this system, there are six modules that manage the human resource system, which are Staff Information/Personnel Management, Leave Management, Claims & Benefit Management, Training Courses, Notice Management and Search module. HRMS provides database-processing system where users do not have to be concerned with the ways in which data are physically stored. Rather, they are free to concentrate on matters important to them.

The main advantage of using this system is that multiple users are allowed to simultaneously access the same system through client (the browser) at anytime and anywhere conveniently. However, as the number of clients increased, the server would quickly become overwhelmed with client requests. Maintenance needs to be carried out regularly to ensure the accuracy, consistency, reliability and timeliness of the data.

Further enhancement on Human Resource Management System is also necessary to fulfill future requirements to increase the effectiveness of the system.

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1:INTRODUCTION

1.1 Project Overview

HRMS (Human Resource Management System) is a computerized, web-based application that enables direct access to human resource information. This system provides online tracking, fast retrieval of data and a paperless environment for information management of an organization. In tradition, all records of employees are managed manually and kept in abundant of files. The number of data, documents and files will increase as the evolution of time. As the consequence, ample space is needed to store the large amount of physical data. Retrieval of these documents for changes of certain data may be difficult if the files are not organized systematically. Furthermore, the files are prone to lost. Hence, HRMS is designed to eliminate the tedium of necessary operational job and reduces the time once required to perform them manually, although users must still input data to this computerized system. HRMS also provides high security of data protection through user authentication. Only valid users are allowed to gain access to this system.

1.2 Features of HRMS

There are various features included in HRMS. Each of the features has included employee-based and administrator-based modules as below:

a) Staff Information/Personnel Management

Personnel Management Module is the core of the HRMS. It manages all particulars related to each employee, either personal or work related information. Some key features of the system includes:

- Maintaining of very comprehensive employee Record.
- Ability to add new employee.
- Update employee's own details or search for certain employee's detail for administrators only.

b) Leave Management

The Leave Management Module performs the function of administering leave for all employees in an organization. Leave entitlement would differ from each employee. Therefore, it is difficult for organizations to manage leave without a proper computerized system. Some key features of the system includes:

- To enable leave application by all level of employees.
- Ability to edit or cancel Leave Application.
- Tracking of Leave Balances (for various type of leave) for each employee.

c) Claims and Benefit Management

Allowances and Benefits form part of the additional perks given to Employees and is sometimes paid out together with Salary Payment or as an Incentive Scheme which is paid separately. The purpose of this module is to identify and

administer all the benefits applicable for employees. Some key features of the system includes:

- Administers all claims, such as Transport Allowance by employees.
- Maintaining employees claims for every month.
- Ability to track and monitor claims limit according to the type of claim and the eligibility of the employee.

d) Training Courses

Organizations today encourage their employees to attend various training courses in order to gain further knowledge and skills about new technology, products, and services. Training Courses module comprises of:

- Ability to view training courses that are offered
- Managing application of training courses

e) Notice Management

This feature allows sending, viewing and deleting notice.

f) Search/Queries

This feature enables the administrators to search for certain employees to view their particulars, including personal and work information. Administrators can delete the employee's account if needed.

1.3 Objectives of HRMS

The main objective of this project is to design, develop and implement an integrated, dynamic, and interactive web-based Human Resource Management System using ASP (Active Server Pages). It consists of a main, attractive homepage for users (employees) to surf and a reliable database system to maintain the human resource information. HRMS uses the Internet as its information transmission medium and the employee can access the information at anytime and anywhere in a secure and convenient environment.

The objectives of HRMS are as follows:

a. **To provide an ease of use, accessibility and user-friendly graphical user interface**

   The design of user interface is at user skill level where no command or programming skill is needed to access the system. Users can handle the system very easily as simple as clicking on buttons only. Users do not have to do a lot of typing; data input is decreased to the minimum. Most of the data are retrieved from the database.

b. To ensure the most reliable and the latest information is distributed in the network

The HRMS information is distributed electronically and most of it is retrieved from the database. Once the database is updated, all the information in the system, as can be seen on the web page, will also be updated automatically.

c. To provide a paperless environment to manage human resource system

Although file-processing systems are a great improvement over manual record-keeping systems, this system is still not reliable because data are often duplicated and incompatible with one another. It is hard to determine which parts of files are needed. The best solution is to use database-processing systems where users are free to concentrate on matters important to them. No paper is needed; all data is stored and retrieved from database.

d. To reduce manpower if possible

By shifting from a manual-paper processing to a computerized processing is able to reduce manpower. Authorized employees can easily manage their own tasks provided in the system electronically instead of depending on one specific employee to cope with all human resource tasks.

1.4 Project Development Methodology

A methodology is a collection of procedures, techniques, tools and documentation aids which helps system developers in their task of implementing a new information system.

A methodology consists of a set of phases, which consist of a set of sub phases. This guides the developers to the choice of techniques at various stages in the project and helps them to plan, manage, control and evaluate information systems project. [1]

There are many types of development model in the software engineering, such as prototyping model, waterfall model, spiral model and others. During the development of this system, the waterfall model is selected as the main approach and prototyping model is used in analysis phase to help users define their information needs and in the design phase to evaluate alternative system configurations.

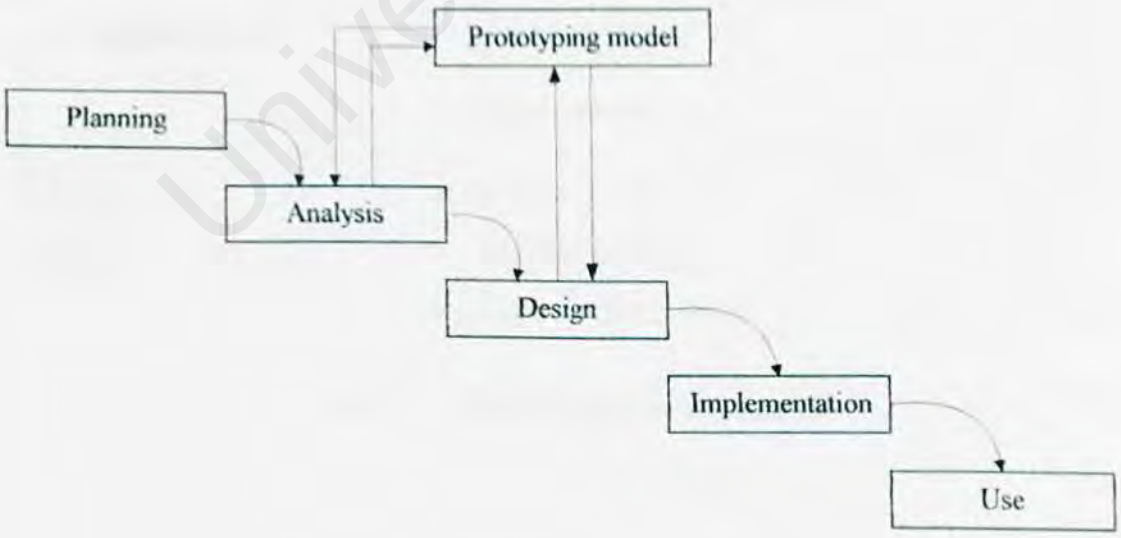

Figure 1.1 Waterfall and Prototyping Model

Waterfall is an approach to development that emphasizes completing a phase of the development before proceeding to the next phase. In conjunction with certain phase completions, a baseline is established that "freezes" the products of the development at that point. If a need is identified to change these products, a formal change process is followed to make the change. Output from each phase includes documentation. [2]

Phase	Activities in phase
1. Planning	❑ Identify the problem ❑ Define the problem ❑ Set system objectives
2. Analysis	❑ Define information needs ❑ Define system performance criteria
3. Design	❑ Prepare the detailed system design ❑ Identify and evaluate alternative system configurations ❑ Select the best configuration ❑ Design interface, database and program
4. Implementation	❑ Code and test system ❑ Install system
5. Use	❑ Use the system ❑ Audit the system ❑ Maintain the system

Table 1.1 Phases in waterfall model

Advantages of using waterfall approach are:

- ✓ System requirements are identified long before programming begins.
- ✓ Minimizes changes to the requirements as the project proceeds.

Disadvantages of this approach are:

- ✓ Design must be completely specified on paper before programming begins.
- ✓ Long time elapses between start and end; sometimes to the point where the requirements change in between as the customer's needs evolve (this causes expensive post programming modifications). [3]

1.5 Project Scope

This project is mainly concerns the development of an online application for the Human Resource department. Advanced tags of HTML such as forms, frames and image will be included in the project to make the website more attractive and easier to navigate. Basically, the HRMS will cover the areas as specified below:

i. implement a password protected website to authorize access for employee only

ii. implement interactive homepages to deploy information to employee and carry out online application process

iii. implement self-management capabilities where an employee has to manage his or her own personal data

iv. develop a database system to keep all records pertaining to the system

v. implement search and queries capabilities for all levels of employee

1.6 Project Limitation

In this project, certain functions are unable to be included due to time constraint and complexity of the module. Below is the limitation:

- ✓ unable to notify the approval of employee's applications by sending e-mail. In my opinion, sending e-mail to employees to notify the approval of their application is much faster and direct to the employee. Furthermore, checking mail is an everyday task, compared to accessing HRMS website which is only when necessary.

1.7 Project Schedule

To achieve the project objective, a project schedule is highly needed and was planned to manage the time and tasks needed to accomplish. A Gantt chart is used to schedule tasks. The project schedule was planned in a set of several milestones as below:

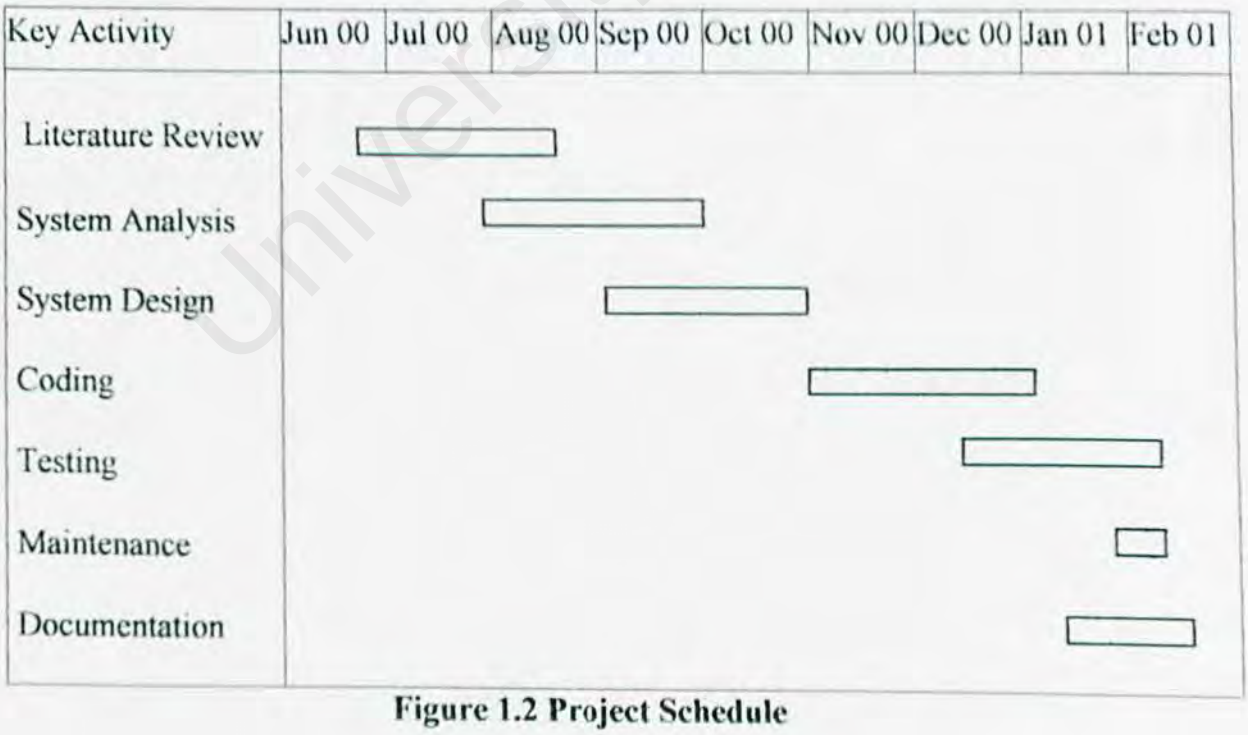

Figure 1.2 Project Schedule

References

[1] http://www.scism.sbu.ac.uk/inmandw/projects/past/9697/projects/rep149/prj149.html

[2] http://www.stsc.hill.af.mil/CrossTalk/1995/jan/Comparis.asp

[3] http://24.72.28.154/372/notes/day2/printerfriendly.html

CHAPTER 2: LITERATURE REVIEW

In the process of developing HRMS, research on the current implemented human resource application has been done. The purpose of this research is to gather more information that will help in developing a more effective Human Resource Management System.

2.1 Project Studies

2.1.1 Definition of Human Resource Management (HRM)

HRM is a term that is now widely used but loosely defined. However, this does not mean that HRM is still in search of a subject area. Beer et al. (1984) in "Readings in HRM" wrote that, "Human Resource Management involves all management decisions and actions that affect the relationship between the organization and employees - its human resources".

Armstrong (1994) in his book HRM: Strategies and Action" defined HRM "as a strategic and coherent approach to the management of an organization's most valued assets - the people working there, who individually and collectively contribute to the achievement of its objectives for sustainable competitive advantage".

Almost every writer who believed that HRM is an approach to manage people effectively agreed that people is the key factor; valued assets in which to invest and sustainable competitive edge can be achieved through them. Organizations can use HRM to gain competitive advantage because it is difficult for competitors to duplicate. [1]

2.1.2 The aims of HRM

Basically, HRM aims to achieve organizational objectives through maximum utilization of the workforce via strategic policies. The aims of HRM are summarized as follows:

- to enable management to achieve organizational objectives through its workforce
- to utilize people to their full capacity and potential
- to foster commitment from individuals to the success of the company through a quality orientation in their performance and that of the whole organization
- to integrate human resource policies with business plans and reinforce an appropriate culture or, as necessary, reshape an inappropriate culture
- to develop a coherent set of personnel and employment policies which jointly reinforce the organization's strategies for matching resources to business needs and improving performance
- to create conditions in which innovation, team working, and total quality can flourish
- to encourage willingness to operate flexibility in the interests of the 'adaptive organization' and the pursuit of excellence".

In a nutshell, it can be said that HRM aims "to place the right people with the right skills in the right jobs for the right cost at the right time". This is supported by the belief that human resources can be the competitive advantage that differentiates an excellent organization from also-rans. [1]

2.2 Web Applications

Open Internet standards have changed the architecture of distributed computing forever. The basic language of the web, HTML, has become a popular language for representing the elements of the user interface. Cross-platform support for scripting languages and Java applets makes it possible to combine dynamic elements with static text for a more interactive user experience. Web technologies are used by organizations to:

- Create an environment for component-based development.
- Enable distribution of applications throughout an enterprise.
- Create and customize new applications quickly.
- Update databases remotely by using an ordinary web browser.
- Add transaction processing to web applications.
- Manage resources and enable remote system administration.
- Provide business-to-business information sharing.

2.3 Client/Server Computing

Before exploring into the details of building a web-based application, it might be helpful to review the architectural model of the Web, and the roles of the browser and server in that model.

Briefly, client/server is a computational architecture that involves client processes requesting service from server processes. In facts, client/server computing is the logical extension of modular programming. Modular programming has as its fundamental assumption that separation of a large piece of software into its

constituent parts ("modules") creates the possibility for easier development and better maintainability. Client/server computing takes this a step farther by recognizing that those modules need not all be executed within the same memory space. With this architecture, the calling module becomes the "client" (that which requests a service), and the called module becomes the "server" (that which provides the service). The logical extension of this is to have clients and servers running on the appropriate hardware and software platforms for their functions. [2]

2.3.1 Characteristics of Client/Server Architecture

The basic characteristics of client/server architectures are:

- combination of a client or front-end portion that interacts with the user, and a server or back-end portion that interacts with the **shared resource**. The client process contains solution-specific logic and provides the interface between the user and the rest of the application system. The server process acts as a software engine that manages shared resources such as databases, printers, modems, or high-powered processors.

- the front-end task and back-end task have fundamentally different requirements for computing resources such as processor speeds, memory, disk speeds and capacities, and input/output devices.

- the environment is typically heterogeneous and **multivendor**. The hardware platform and operating system of client and server are not usually the same.

Client and server processes communicate through a well-defined set of standard application program interfaces (API's) and RPC's.

- an important characteristic of client-server systems is **scalability**. They can be scaled horizontally or vertically. Horizontal scaling means adding or removing client workstations with only a slight performance impact. Vertical scaling means migrating to a larger and faster server machine or multiserver. [3]

2.3.2 Two-Tier Architecture

A two-tier architecture is where a client talks directly to a server, with no intervening server. It is typically used in small environments (less than 50 users).

Early two-tier (client/server) applications were developed to access large databases, and manipulate the data with the user interface into the client application. The server's task was simply to process as many requests for data storage and retrieval as possible.

Two-tier applications perform many of the functions of stand-alone systems: They present a user interface, gather and process user input, perform the requested processing, and report the status of the request. This sequence of commands can be repeated as many times as necessary. Because servers provide only access to the data, the client uses its local resources to perform most of the processing.

One major advantage of this model was that by allowing multiple users to simultaneously access the same application data, updates from one computer were

instantly made available to all computers that had access to the server. However, this approach will usually result in an ineffective system, as the server becomes overwhelmed as the number of clients increased. To properly scale to hundreds or thousands of users, it is usually necessary to move to a three-tier architecture.

A two-tier client/server model has several critical limitations:

- **Not scalable**. The inability of a two-tier approach to grow beyond the physical boundaries of a client machine and a server machine prevents this model from being scalable.
- **Unmanageable**. Because we cannot encapsulate business rules and deploy them centrally, sharing common processes and reusing our work is difficult at best.
- **Poor performance**. The binding of the graphical interface to the data source consumes major resources on the client machine, which results in poor performance and, unfortunately, unhappy clients. [4]

2.3.3 Three-Tier Architecture

A three-tier architecture is a multitier application, also known as *n-tier* architecture. It introduces a server (or an "agent") between the client and the server, the business logic. The role of the agent is manifold. It can provide translation services (as in adapting a legacy application on a mainframe to a client/server environment), metering services (as in acting as a transaction monitor to limit the number of simultaneous requests to a given server), or intelligent agent services (as in mapping a

request to a number of different servers, gathering the results, and returning a single response to the client.

Most systems will perform the following three main tasks, which correspond to three tiers, or layers, of the *n*-tier model:

Task	**Description**
User interface and navigation	Labeled Tier 1 in the following graphic, this layer comprises the entire user experience. Not only does this layer provide a graphical interface so that users can interact with the application, input data, and view the results of requests, it also manages the manipulation and formatting of data once the client receives it. In web applications, the browser performs the tasks of this layer.
Business logic	Tier 2, between the interface and data services layers, is the domain of the distributed application developer. Business logic, which captures the rules that govern application processing, connects the user at one end with the data at the other. The functions that the rules govern closely mimic everyday business tasks, and can be a single task or a series of tasks.

Data services	Shown as Tier 3 in the following graphic, data services are provided by a structured (SQL, Oracle database) or unstructured (Microsoft Exchange, Microsoft Message Queuing) data store, which manages and provides access to the application data. A single application may enlist the services of one or more data stores.

Table 2.1 Three-Tier Architecture Tasks

This model requires much more analysis and design up front, but greatly reduces maintenance costs and increases functional flexibility in the long run. The following diagram depicts the Microsoft technologies that service the various tiers in the new system design. [5]

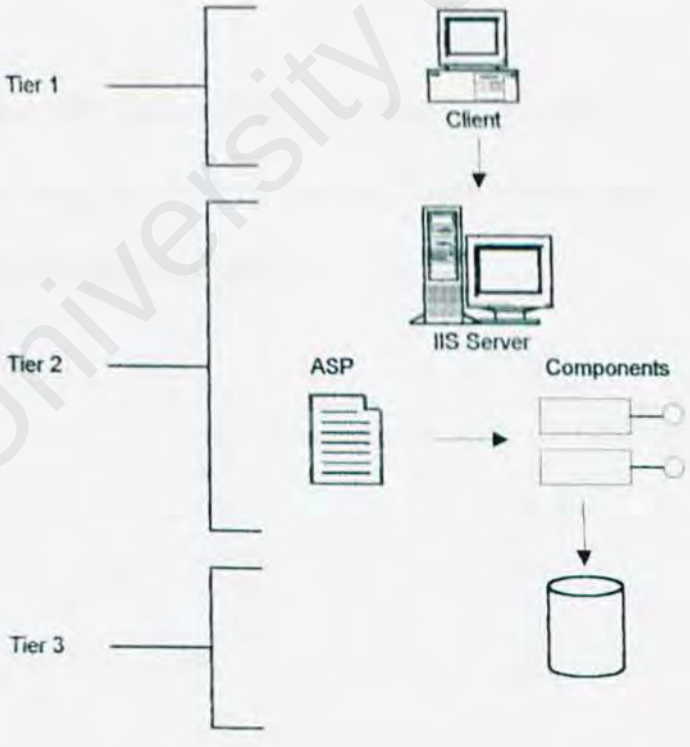

Figure 2.1 Three-Tier Architecture Diagram

The following benefits illustrate the value of distributed three-tier client/server development:

- **Reuse**. The time of investing in designing and implementing components is not wasted because applications can be shared.
- **Performance**. Because the components on machines can be deployed other than the client workstation, we have the ability to shift processing load from a client machine that might be underpowered to a server with extra horsepower. This flexibility in deployment and design enables developers to take advantage of the best possible methods for each aspect of the application's execution, and results in better performance.
- **Manageability**. Encapsulation of application's services into components enables large, complex applications to be broken down into more manageable pieces.
- **Maintenance**. The centralization of components for reuse has an added benefit. They become easier to redeploy when modifications are made, thus keeping pace with business needs. [4]

2.4 Tools and Technologies Consideration

This session will examine the solutions being used to develop the project and the alternative solutions, which could have used to achieve the same results.

2.4.1 Development Platform

2.4.1.1 Windows NT Server 4.0

Microsoft Windows NT Server 4.0 is a true multipurpose server operating system. A multipurpose operating system does more for less because it replaces or integrates several disparate, single-purpose servers to run various applications, which is an efficiency that helps simplify and consolidate the computing environment. The services it provides are designed to address customer requirements in every category. Windows NT Server includes support for applications, Web services, communications, and much more.

Following are the list of features in Windows NT Server 4.0:

1) Web Services

The combination of web and operating system services in Windows NT Server 4.0 makes it possible for the first time to deploy scalable and reliable web-based applications. Windows NT Server 4.0 incorporates Internet Information Server 4.0 (IIS) services, enhancing the operating system into a singular, cohesive platform that combines easy, reliable applications services with comprehensive web services.

a) Internet Information Server 4.0

IIS supports multiple web server scenarios, ranging from simple web sites on a corporate intranet to large Internet Service Provider (ISP) web-hosting firms. Specifically, IIS 4.0 accommodates multiple web sites with support for host headers, web site operators, per-web site bandwidth throttling, and enhanced web-based administration. IIS 4.0 provides a number of features that make it easier to build dynamic web sites.

b) Crystal Reports

Crystal Reports, introduced with Service Pack 3, is a client/server report writer used to create presentation-quality reports and integrate them into database applications. Crystal Reports for IIS includes features to create reports from web server log files and includes preformatted web log reports. HTML publishing capabilities instantly turn reports into presentation-quality, information-rich web documents.

2) Communications Services

Windows NT Server 4.0 provides the easy and cost-effective way to connect branch offices, remote employees and the internet. Advanced networking capabilities are provided by its services such as routing and Remote Access Services (RAS). These services, which don't require changes in client software, provide an easy, inexpensive way to deploy powerful Virtual Private Networks (VPNs) for securely sharing information over the Internet. The communications services for Windows NT Server

4.0 have been further improved with the addition of Internet Connection Services for RAS 1.0.

a) Internet Connection Services for RAS 1.0

Internet Connection Services for RAS (ICS) can significantly reduce remote connectivity costs and improve end-user experience. ICS can help implement an economical communication infrastructure that takes advantage of the internet and Internet Service Provider (ISP) services to reduce remote access costs. These technologies can also be used to lower the total costs of remote access services by outsourcing remote access hardware, administration, and support to an ISP.

3) Application Services and Tools

In addition to its Web and communications services, Windows NT Server 4.0 is designed to be an application server with the base services application developers need, such as virtual memory and multi-tasking, built in to the operating system.

a) Microsoft Transaction Server (MTS) 2.0

MTS 2.0 is a component-based transaction processing system that enables the development, deployment and management of high-performance, scalable, and robust server applications. MTS 2.0 simplifies the development and deployment of server-centric applications built using Microsoft Component Object Model (COM) technologies.

MTS 2.0 is the easiest way to build and deploy COM-based applications. It offers comprehensive component functionality, and it is fully integrated with

other features of Windows NT Server 4.0features, such as IIS 4.0 and Active Server Pages for easy Internet or intranet application development.

4) Security

Comprehensive, usable security enables Windows NT Server 4.0 to offer the most robust security model of any server or workstation operating system on the market. The centralized Windows NT security subsystem uses advanced security design features that provide an exceptional level of system security.

To further strengthen security access and add flexibility, Windows NT Server 4.0 has been enhanced with ever more usable features.

a) Password Filtering

A password filter (PASSfilt.dll) allows system administrators to increase password strength. A part of Service Pack 3 and later Service Packs, this filter is copied to %system root %\SYSTEM32 when the Service Pack is installed onto the system. The password filter should be copied to the primary domain controller for the domain, and to any backup domain controllers in the event the server role in the domain changes.

b) Restricting anonymous user access

Windows NT has a feature that allows users to list domain user names and enumerate share names. For administrators who want to restrict access to this for anonymous logon users (also known as NULL session connections), Microsoft added a mechanism in Service Pack 3. Service Pack 3 also

introduced a feature that restricts anonymous logon users from connecting to the registry remotely. After Service Pack 3 or a later Service Pack is installed, anonymous users cannot connect to the registry and cannot read or write to any registry. [6]

2.4.1.2 Windows 2000 Server

Windows 2000 Server, aimed at small-to-medium size businesses. It can function as a Web server or a workgroup (or branch office) server. It can be part of a two-way symmetric multiprocessing system.

Early reviews of Windows 2000 report that it is more stable (less apt to crash) than Windows 98/NT systems. A significant new feature is Microsoft's Active Directory, which, among other capabilities, enables a company to set up virtual private network, to encryption data locally or on the network, and to give users access to shared files in a consistent way from any network computer.

Notable features of the Windows 2000 products are:

- A fully-customizable administrative console that can be based on tasks rather than files, applications, or users
- A new file directory approach called Active Directory that lets the administrator and other users view every file and application in the network from a single point-of-view.
- Dynamic Domain Name Server (DNS), which replicates changes in the network using the Active Directory Services, the Dynamic Host Configuration Protocol (Dynamic Host Configuration Protocol), and the Windows Internet

Naming Service (Windows Internet Naming Service) whenever a client is reconfigured.

- The ability to create, extend, or mirror a disk volume without having to shut down the system and to back up data to a variety of magnetic and optical storage media.
- A Distributed File System (DFS) that lets users see a distributed set of files in a single file structure across departments, divisions, or an entire enterprise.
- Close integration with and support for Microsoft's Message Queue Server, Microsoft Transaction Server, and Internet Information Server (Internet Information Server). [7]

2.4.1.3 Windows 98 Second Edition

Windows 98 is a widely installed product in Microsoft's evolution of the Windows operating system for personal computers. Designed for consumers, Windows 98 allows personal computer to work better and play better. Building on the features of Windows 98, Windows 98 Second Edition improves the online experience, enhances hardware support, and delivers home networking capabilities.

Windows 98 also provides a 32-bit file allocation table that allows having a single-partition disk drive larger than 2 Gigabytes. Other features in Windows 98 include:

- **Better performance and reliability**

  Microsoft Windows 98 improves on key areas that keep users waiting on their PCs today, including opening applications an average of 36 percent faster, rendering Internet pages up to 25 percent faster and shutting down

the PC up to two to five times more quickly than with Windows 95. In addition, Windows 98 can provide users an average of 28 percent more disk space through more efficient storage of data on the hard drive.

- **Improved ease of use and Internet access**

  Windows 98 enhanced Internet browsing functionality with Internet Explorer 5 includes ease-of-use and personalization innovations such as Explorer bars, improvements in security, and new search capabilities. Conferencing and multimedia technologies in Windows NetMeeting 3 and Windows Media Player 6.1 provide a faster and easier way to take advantage of all that the Internet has to offer.

  Dial-up networking improvements provide users with faster remote network and Internet connections as well as an automated connection to online services.

- **A new generation of hardware and entertainment**

  Just as Windows 95 enabled a range of new software applications, Windows 98 SE empowers a new range of hardware and entertainment functionality. Windows 98 SE includes native support for Universal Serial Bus (USB), which makes setting up additional hardware as easy as plugging in a toaster. Windows 98 also provides DVD and television reception capabilities, allowing a PC with a TV tuner card to seamlessly receive and display television and other data distributed over broadcast networks.

Windows 98 is gradually being replaced by Windows 2000 that are designed for personal or small-office professional or business use. [8]

2.4.2 Database Management System

A database is basically a computerized record keeping system with the purpose of maintaining the data stored within and, making it available on demand.

Apart from a database solution, a conventional application based system could be used in terms of data storage and retrieval. However, the former was chosen because of the advantages the database has over an application based systems for the following reasons.

- **Database Management System (DBMS)**

  A major advantage of the database approach is the ability of the DBMS (software to manage data) to provide logical data independence between applications. The database concept moves the basis of the data file away from individual application structures into an integrated collection of data largely independent of the application using it.

- **Reduction In Data Redundancy**

  Due to the sharing of files in a database, data redundancy can be kept at a minimum. For example, a personnel and production department may both use a file that includes employee details. These two files can be combined and integrated together and the redundancy illuminated.

- **Inconsistency**

  Reduced redundancy also implies reduced inconsistency, since there is less scope for discrepancies within a single database. This should improve the overall quality of the decision-making and management.

- **Security**

  Having complete control over the database should increase the security of the system. The only means of access to the database should be through proper secured channels. These security rules can be checked whenever an access is made.

- **Data Sharing**

  Sharing does not have to be restricted to existing application in the database, but also the new application can be developed to operate against the same shared data; it may be possible to satisfy requirements of new applications without having to create any additional stored data. [9]

There are many criteria that go into choosing a database. For example:

- Capacity – can the database support the amount of information that we want to store?
- Scalability – can it support the number of simultaneous users we are expecting?
- Cost – how much does the database system cost?
- Administration – how hard or easy is it to administer the system?
- Compatibility – does the database work with our other system?

- Performance – how many queries can the database engine execute in the given time?

2.4.2.1 Microsoft Access 2000

Microsoft Access is a relational database management system to create relational database. Together with the ODBC driver, data can be retrieved from the database in client/server system.

The Microsoft Access database uses the Microsoft Jet database engine. The Jet database tables for a single database are stored in a single data file, usually with the extension ".mdb". The general way of accessing this database is through an ODBC driver or OLE-DB provider, via ADO.

However, the Jet engine is not able to support more than 5-10 simultaneous users; moreover, there are problems inherent in relying on single file to store all database information. It also does not support a resource manager interface – therefore it is incapable of managing durable and large amount of data during a transaction. [10]

2.4.2.2 Microsoft SQL Server 7.0

Microsoft SQL Server is a relational database engine that is available for the Windows NT platform. It is a true database server system – as opposed to the file-based system used by the Jet database. This gives it vastly increased power, scalability and robustness advantage over Jet.

The biggest advantage of using SQL Server is that it supports access to its data via a resource manager. The resource manager will support the maintenance of durable data

– which means that any changes made to the database system during a transaction will be automatically reversed if the transaction does not complete successfully.

Overall, the benefits of SQL Server 7.0 are:

- ✓ Reliable distributed data and transaction
- ✓ Centralized control of distributed servers
- ✓ Very high performance and scalability
- ✓ Support for large databases
- ✓ Full programmability and standards support
- ✓ Rich desktop integration

2.4.3 Microsoft Data Access Technology

2.4.3.1 Open Database Connectivity (ODBC)

ODBC is a standard for accessing data. It was designed to allow the programmers to use a common set of routines to access the data stored in database, regardless of the type of database in which the data is stored. They can manipulate the data without worrying the exactly where the data was stored, or which type of database was storing it. It provided interface transparency – the programmer can access an Oracle database in the same way they accessed a SQL Server database.

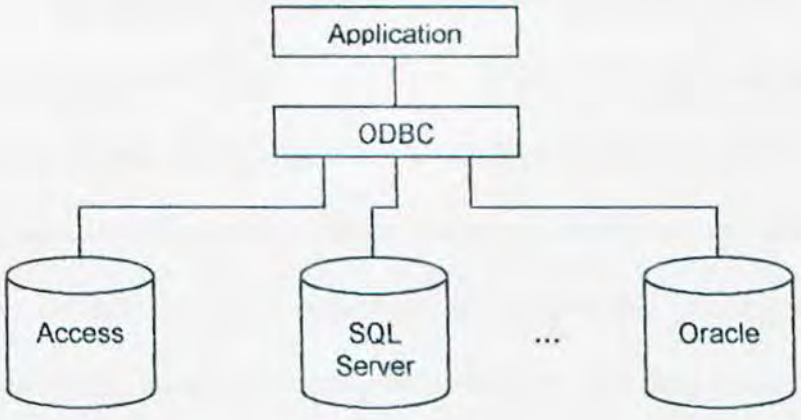

Figure 2.2 ODBC Data Access

Of all the methods for accessing data, the Open Database Connectivity (ODBC) is the most efficient in terms of execution speed. In terms of programming, it requires the most time and the most caution. [10]

2.4.3.2 OLE-DB

OLE-DB is faster and easier to use than ODBC. The following diagrams is a picture of data access using OLE-DB:

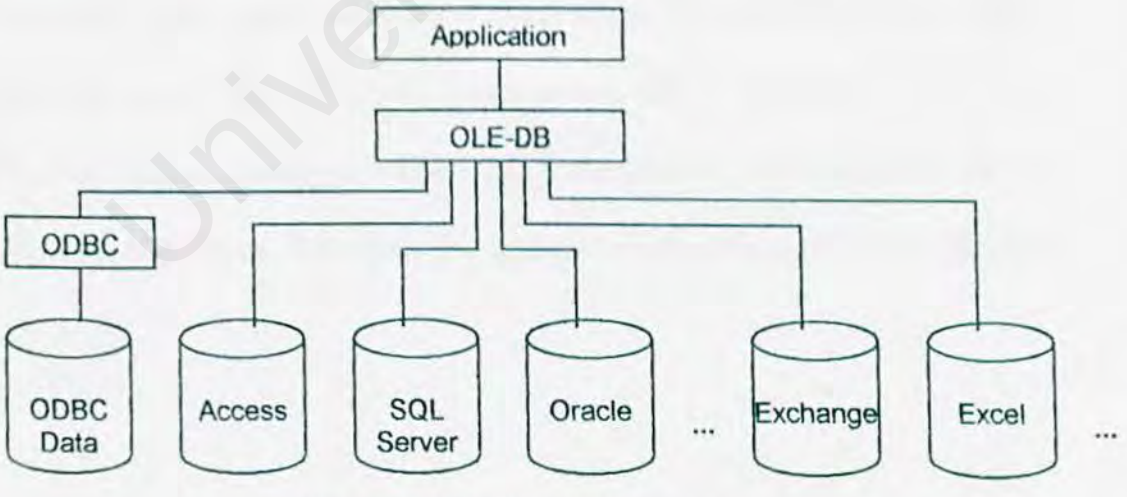

Figure 2.3 OLE-DB Data Access

As can be seen from the diagram, OLE-DB is very similar to idea of ODBC, but it allows access to a much broader range of data stores. In fact, OLE-DB even supports database connections through ODBC, so that effectively the generic OLE-DB layer allows the connection to legacy database through existing ODBC connections. OLE-DB introduces data providers and data consumers. A data provider is something that provides data, and the data consumer is something that uses the data. OLE-DB data providers provide data for data consumers, such as an application written in language like Visual Basic or Visual C++. [10]

2.4.3.3 ActiveX Data Objects (ADO)

ADO (ActiveX Data Objects) is an application program interface (API) from Microsoft that lets programmers writing Windows applications get access to relational and nonrelational databases from both Microsoft and other database providers.

Like Microsoft's other system interfaces, ADO is an object-oriented interface. ADO is a higher-level model than OLE-DB, which means that it simplifies some of the complexities of programming with OLE-DB. Thus, ADO is much easier to use than OLE-DB. The ADO layer is between the applications itself and the OLE-DB layer. [10]

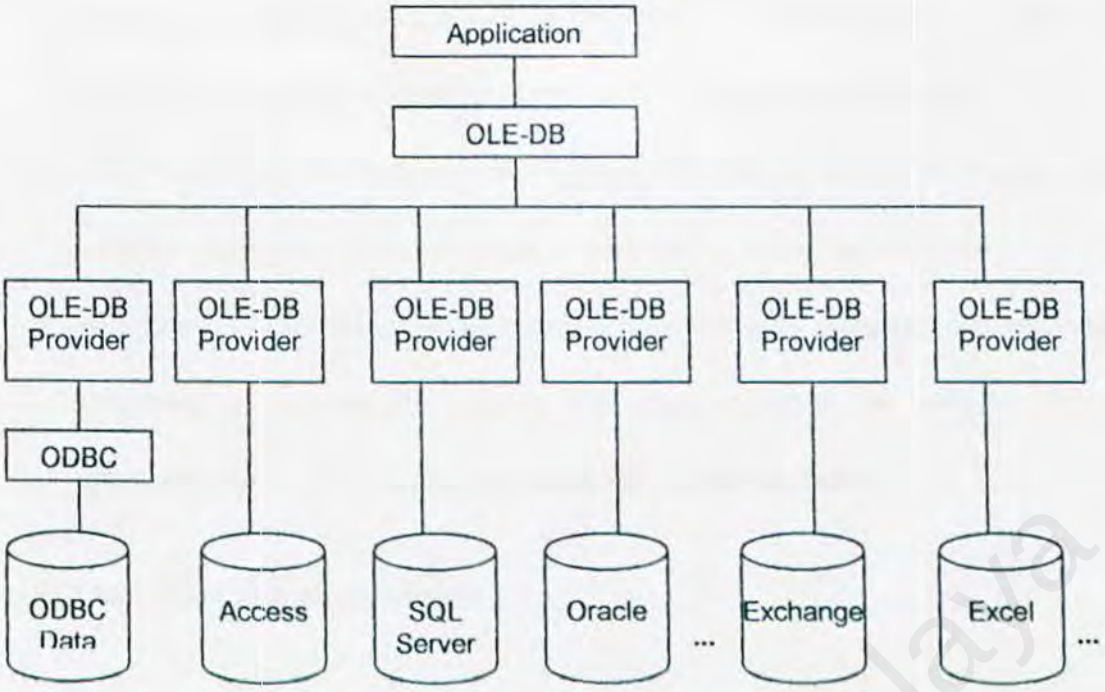

Figure 2.4 ADO Data Access

2.4.4 Web Development Technology

2.4.4.1 Internet Information Server 5.0 (IIS 5.0)

Internet Information Server (IIS) 4.0 is a built-in Web server in Microsoft Windows NT Server 4.0 and Windows 2000 that provides the easiest way to publish information and bring HRMS to the Web.

The advantages of IIS 5.0 include:

Integration with Windows NT Server

- IIS provides the highest performance of any Web server on Windows NT Server and delivers superior price-performance to comparable UNIX-based systems. In

addition, the application services of Windows NT Server provide a reliable and scalable platform for building and deploying Web-based applications.

- Only IIS brings the security of Windows NT Server to the Web site, without addition configuration, to protect all the human resources information.
- Windows NT Server combined with IIS is the only solution that provides an integrated set of tools for running and managing all of the network, Web and application services, reducing training time for administrators.

Publish and share information easily

- Create professional-quality Web pages and publish entire sites, without knowing any HTML, by using new wizards and templates.
- Publish information to the Web quickly and easily using a Web browser, the Web publishing wizard or FTP.
- Share files and data on Windows NT, Novell NetWare and UNIX servers and more than 55 databases, including Microsoft SQL Server, Oracle and Sybase databases.
- Search securely and easily for content in HTML and Microsoft Office document types and multiple languages.

Build and run web applications

- Protect applications and Web sites against failure from misbehaving components or Web applications on the server, by running them in separate memory spaces, a feature known as process isolation.
- Rapidly create scalable Web applications using built-in distributed application

services that automatically scale to serve thousands of simultaneous users. Only Windows NT Server with IIS provides this built-in capability.

- Developers familiar with desktop applications written in any language can now write and easily debug components and applications that run on the server.
- Integrated Java Virtual Machine can be used to provide a reliable and high-performance environment for running Java components on the server with ASP pages.

Microsoft BackOffice and Internet Explorer

As the only Web server that is an integrated with Windows NT Server, Windows NT Server with IIS provides the easiest way to bring the robust capabilities of the Microsoft BackOffice suite of server applications to the Web. With IIS, we are able to take advantage of:

- Powerful Microsoft SQL Server database.
- Web site life-cycle management capabilities of Microsoft Site Server.
- Messaging and groupware capabilities of Microsoft Exchange Server.
- Host and data connectivity of Microsoft SNA Server. Combine BackOffice, Windows NT Server with IIS, and Microsoft Internet Explorer and you have an integrated client/server solution for sharing information and running applications on corporate intranets and the Internet. [11]

2.4.4.2 Common Gateway Interface (CGI)

The Common Gateway Interface (CGI) is a mechanism for creating scripts on the server, which can then be used to create dynamic web applications. CGI is the standard way for a Web server to pass a Web user's request to an application program and to receive data back to forward to the user. When the user requests a Web page (for example, by clicking on a highlighted word or entering a Web site address), the server sends back the requested page. However, when a user fills out a form on a Web page and sends it in, it usually needs to be processed by an application program. The Web server typically passes the form information to a small application program that processes the data and may send back a confirmation message. This method or convention for passing data back and forth between the server and the application is called the *common gateway interface (CGI)*. It is part of the Web's HTTP protocol.

However, CGI has some severe shortcomings. The major one is that it adds an extra level to our browser-server model of interaction: namely, it is necessary to run a CGI program to create the dynamic page, before the page is processed on the server. Also, the format in which CGI receives and transmit data means that this data is not easily manipulated by many programming languages, hence programming languages that has good facilities for manipulating text and communicating with other software has to be used. The most able programming languages that can work on any operating system for doing this are C, C++ and Perl. Visual Basic does not offer sufficiently adequate text-handling facilities, and is therefore rarely used with CGI. [10]

2.4.4.3 ASP

Active Server Pages, or ASP, is the programming model for Microsoft Internet Information Server (IIS). ASP enables developers to build rich Web applications quickly by combining programming logic (written in the script language of their choice) with HTML pages. ASP pages are typically used to provide security for a Web site, access databases, and call business logic encapsulated in COM objects. The ASP engine on the server compiles and runs the script. The browser receives only the resulting HTML pages that are constructed from the server-side script. ASP script can be programmed in either JScript or VBScript. ASP files can also created by using ActiveX Data Objects (ADO) program statements in the HTML file.

An ASP is somewhat similar to a server-side include or a common gateway interface (CGI) application in that all involve programs that run on the server, usually tailoring a page for the user. Typically, the script in the Web page at the server uses input received as the result of the user's request for the page to access data from a database and then builds or customizes the page on the fly before sending it to the requestor.

As a conclusion, Active Server Pages (ASP) is a server-side scripting environment that can be used to create and run dynamic, interactive Web server applications. With ASP, we can combine HTML pages, script commands and COM Components to create powerful Web-based applications, which are easy to develop and modify.

2.4.4.4 Comparison between ASP and CGI

Active Server Pages (ASP) provides all of the functionality of CGI applications in an easier-to-use and more robust environment.

ASP is an easier way for your server to access information in a form not readable by the client (such as an SQL database) and then act as a gateway between the two to produce information that the client can view and use.

With CGI, the server creates as many processes as the number of client requests received. The more concurrent requests there are, the more concurrent processes created by the server. However, creating a process for every request is time-consuming and requires large amounts of server RAM. In addition, this can restrict the resources available for sharing from the server application itself, slowing down performance and increasing wait times on the Web.

ASP runs in the same process as the Web Server, handling client requests faster and more efficiently. It is much easier to develop dynamic content and Web applications with ASP. [12]

2.4.4.5 JavaScript

JavaScript is an interpreted programming or script language from Netscape. It is somewhat similar in capability to Microsoft's Visual Basic, Sun's Tcl, the UNIX-derived Perl, and IBM's REXX. In general, script languages are easier and faster to

code in than the more structured and compiled languages such as C and C++. Script languages generally take longer to process than compiled languages, but are very useful for shorter programs.

JavaScript is used in Web site development to do such things as:

- Automatically change a formatted date on a Web page
- Cause a linked-to page to appear in a popup window
- Cause text or a graphic image to change during a mouse rollover

JavaScript uses some of the same ideas found in Java, the compiled object-oriented language derived from C++. JavaScript code can be imbedded in HTML pages and interpreted by the Web browser (or client). JavaScript can also be run at the server as in Microsoft's Active Server Pages (ASPs) before the page is sent to the requestor. Both Microsoft and Netscape browsers support JavaScript, but sometimes in slightly different ways. [13]

2.4.4.6 VBScript

VBScript is an interpreted script language from Microsoft that is a subset of its Visual Basic programming language.

In general, script languages are easier and faster to code in than the more structured, compiled languages such as C and C++ and are ideal for smaller programs of limited capability or that can reuse and tie together existing compiled programs.

VBScript is Microsoft's answer to Netscape's popular JavaScript. Both are designed to work with an interpreter that comes with a Web browser - that is, at the user or client end of the Web client/server session. VBScript is designed for use with Microsoft's Internet Explorer browser together with other programming that can be run at the client, including ActiveX controls, automation servers, and Java applets. Although Microsoft does support Netscape's JavaScript (it converts it into its own JScript), Netscape does not support VBScript. For this reason, VBScript is best used for intranet Web sites that use the Internet Explorer browser only. [14]

2.4.4.7 HTML

HTML (Hypertext Markup Language) is the set of "markup" symbols or codes inserted in a file intended for display on a World Wide Web browser. The markup tells the Web browser how to display a Web page's words and images for the user. The individual markup codes are referred to as elements (but many people also refer to them as tag).

HTML is a standard recommended by the World Wide Web Consortium (W3C) and adhered to by the major browsers, Microsoft's Internet Explorer and Netscape's Navigator, which also provide some additional non-standard codes. The current version of HTML is HTML 4.0. However, both Internet Explorer and Netscape implement some features differently and provide non-standard extensions. Web developers using the more advanced features of HTML 4.0 may have to design pages for both browsers and send out the appropriate version to a user. Significant features in HTML 4.0 are sometimes described in general as dynamic HTML. What is

sometimes referred to as HTML 5.0 is an extensible form of HTML called Extensible Hypertext Markup Language. [15]

2.4.5 Web Applications Tools

2.4.5.1 Microsoft Visual Interdev 6.0

Visual Interdev comes as a part of Microsoft's suite of professional programming tools, known as Visual Studio. It is a tool for designing dynamic web applications. Support for Active Server Pages, middle-tier components written in any language, Dynamic HTML, and integrated database design and programming tools make Visual Interdev the ideal tool for building dynamic, data-driven Web sites.

Visual InterDev 6.0 enables developers to build applications accessible from any platform running a standard Web browser such as Microsoft Internet Explorer or Netscape Navigator. The Visual InterDev development environment itself runs on Microsoft Windows 98 or Microsoft Windows NT 4.0 or later.

Visual InterDev offers a user interface similar to those for Visual Basic, Visual J++, and Visual Studio. Using Visual InterDev, one can assemble pages that use Microsoft's ActiveX technologies, including Active Server Page (Active Server Page) technology. The developer can build and insert ActiveX control or Java applet. Visual InterDev includes an HTML editor and support for dynamic HTML. The Web site can be integrated with server programs written in any language and access to almost any database using Microsoft's Universal Data Access, including ActiveX Data Objects, Open Database Connectivity, and OLE DB.

One simple but very useful feature of VI 6.0 is that it highlights ASP <% and %> tags in yellow, and the ASP script itself is highlighted using blue for legal keywords – so they stand out from the HTML.

In addition, VI boasts strong links with SQL Server, which makes it very easy to set up database combining ASP and SQL Server. It also provides several useful web-based tools for checking links, highlighting the broken ones on your site,, and allowing to drag-and-drop pages from one location to another.

2.4.5.1.1 Difference between Microsoft FrontPage and Microsoft Visual InterDev

Visual InterDev is a Web development tool designed for programmers, while Microsoft FrontPage is a Web authoring tool designed for non-programmers. Microsoft FrontPage is a member of the Microsoft Office family, and looks and works like other Office applications. Visual InterDev is a member of the Microsoft Visual Tools family, and looks and works like other Microsoft visual development tools, including Microsoft Visual C++, Microsoft Visual J++, Microsoft Visual FoxPro and Microsoft Visual Basic.

Because most Web sites are created by teams of people, including both programmers and non-programmers, Visual InterDev and Microsoft FrontPage interoperate so that teams of people with different sets of skills can work together on the same Web site. Webmasters, technical developers, database administrators, and MIS personnel would require Visual InterDev. On the other hand, content publishers - such as sales, marketing, graphic arts, and finance - would need FrontPage. [16]

2.4.5.2 Microsoft Visual Basic 6.0

Microsoft Visual Basic 6.0 is a member of the Microsoft Visual Studio development tools suite. It is the most productive tool for creating high-performance components and applications. Visual Basic 6.0 offers developers the ability to create robust applications that reside on the client or server, or operate in a distributed n-tier environment. Visual Basic 6.0 is the Rapid Application Development (RAD) tool available either as a stand-alone product or as a part of the Visual Studio 6.0 suite of tools.

Key features include the following:

- **Integrated Visual Database Tools.**

  Visual Basic 6.0 provides a complete set of tools for integrating databases with any application. Database features include design tools for creating and modifying Microsoft SQL Server, Oracle 7.3.3 or above, and AS/400 databases.

- **Data Environment Designer.**

  Visually create reusable recordset command objects with drag-and-drop functionality. Bind to multiple data sources for data aggregation and manipulation.

- **Drag-and-drop creation of data-bound forms and reports.**

  Using the new Data Environment designer, developers can quickly drag-and-drop custom data-bound controls to create forms; they can create reports with the new Data Report Designer. Creation of custom data hierarchies is as easy as filling out a dialog box and dragging the command to the form.

- **Visual Basic WebClass Designer.**

  Create server-side applications and components that are easily accessible from any Web browser on any platform.

- **Dynamic HTML Page Designer.**

  Develop multimedia-rich applications using the document object model and Dynamic HTML surface as the user interface.

- **New performance enhancements.**

  Use new String Functions and features such as "Retain in Memory" to create faster, more scalable applications and components. [17]

2.4.6 Others

2.4.6.1 ActiveX

ActiveX is the name Microsoft has given to a set of "strategic" object-oriented program technologies and tools. The main technology is the Component Object Model (COM). Used in a network with a directory and additional support, COM becomes the Distributed Component Object Model (DCOM). The main thing that we

create when writing a program to run in the ActiveX environment is a component, a self-sufficient program that can be run anywhere in the ActiveX network (currently a network consisting of Windows and Macintosh systems). This component is known as an ActiveX control. ActiveX is Microsoft's answer to the Java technology from Sun Microsystems. An ActiveX control is roughly equivalent to a Java applet.

One of the main advantages of a component is that it can be re-used by many applications (referred to as component containers). A COM component object (ActiveX control) can be created using one of several languages or development tools, including C++ and Visual Basic, or PowerBuilder, or with scripting tools such as VBScript.

Currently, ActiveX controls run in Windows 95/98/NT and in Macintosh. [18]

2.4.6.2 ActiveX DLL

An ActiveX DLL is s code component that runs in the same process as the client applications. Thus, it runs in the same address space as the client application that is using the component.

Any component created to work with the Microsoft Transaction Server must be ActiveX DLL component. Any other components developed for use on the same machine as client application can also be ActiveX DLL components. This is normally the preferred choice because it is easier to test and has better performance.

The advantages of using ActiveX DLLs are:

- Code can be easily shared among applications.

- They offer excellent performance due to the in-process nature of the component.
- Fixing a bug in DLL Implement object only requires distributing an updated DLL. All applications using the DLL are immediately fixed.
- Any OLE automation client, including all VBA-based applications (such as Microsoft Office) and other Windows development languages can use them.

The disadvantages are:

- If an updated DLL is incompatible with its predecessor, every application that uses the DLL can be broken.
- It does not support multithreaded objects in Visual Basic 5.0.
- It increases the complexity of deploying an application.
- It requires registration, version checking and component verification for safe distribution.

References

[1] http://adtimes.nstp.com.my/jobstory/articles.com

[2] http://msdn.microsoft.com/library/psdk/iisref/eadg4cbp.htm

[3] http://www.faqs.org/faqs/client-server-faq/section-4.html

[4] http://www.vbhow.to/books/vb6cs/ch01/ch01.asp

[5] http://msdn.microsoft.com/library/psdk/iisref/eadg6dv2.htm

[6] http://www.microsoft.com/ntserver/nts/default.asp

[7] http://www.whatis.com/WhatIs_Definition_Page/0,4152,213368,00.html

[8] http:// www.microsoft.com/windows98/guide/Win98/MoreInfo/FactSheet.asp

[9] http://www.scism.sbu.ac.uk/inmandw/projects/past/9697/projects/ rep149/ prj149.html

[10] David Buser, John Kauffman, Juan T.Llibre, Brian Francis, David Sussman, Chris Ullman, Jon Duckett, Beginning Active Server Pages 3.0, Wrox Press Ltd., 1999

[11] http://www.microsoft.com/ntserver/iis/default.asp

[12] http://msdn.microsoft.com/aspfaq.htm

[13] http://www.whatis.com/javascript.htm

[14] http://www.whatis.com/vbscript.htm

[15] http://www.whatis.com/WhatIs_Definition_Page/0,4152,212286,00.html

[16] http://msdn.microsoft.com/vinterdev/prodinfo/qa.htm

[17] http://msdn.microsoft.com/vbasic/prodinfo/qa.htm

[18] http://www.whatis.com/activex.htm

CHAPTER 3: METHODOLOGY AND ANALYSIS

3.1 Methodology

As mention in Chapter 1, the methodology that has been introduced in developing this project is the waterfall model. This methodology is important to ensure the project is well planned from the beginning stage until the end of the project. From research, waterfall model consists of the following phases:

- Planning
- Analysis
- Design
- Implementation
- Use

The Planning Phase

✓ Benefit from planning the project

- **Define the scope of the project.** Which activities or systems will involve? Which will not? This information provides an initial estimate of the scale of resources required.
- **Recognize potential problem areas.** Planning will point out things that might go wrong so they may be prevented.
- **Arrange a sequence of tasks.** Many separate tasks will be necessary to achieve the system. These tasks are arranged in a logical sequence based on information priorities and the need for efficiency. [1]

✓ Steps of Planning Phase

- **Identify the problem.** This phase requires that the analyst look honestly at what is occurring. Then, together with other organizational members, the analyst pinpoints problems.
- **Define the problem.** Once they realize that a problem exists, he or she must understand it well to pursue a solution. This may includes identifying where the problem exists and its cause.
- **Set system objectives.** The analyst develops a list of objectives that the system must meet in order to satisfy the users.

The Analysis Phase

✓ With the planning completed, the project teams turns to analysis of existing system. System analysis is the study of an existing system for the purpose of designing a new or improved system.

✓ During the analysis phase:

- **Define information needs.** Analysts learn about the user's information needs by engaging in a variety of information-gathering activities, including personal interviews, observations, record searches and surveys. Analyst assembles and reviews the documentation of existing system.

- **Define system performance criteria.** When information needs are defined, exact terms what the system should accomplish – its performance criteria is specified.

The Design Phase

- ✓ With the understanding of the existing system and the requirement for the new system, the design of the new system can be addressed. System design is the determination of the processes and data that are required by a new system.
- ✓ The steps of the design phase are:
  - **Prepare the detailed system design.** Model solution process (Data Flow Diagram) and Model solution data structure (Entity-Relationship diagram) are designed to chart the input, processes, output and the relationships of the system's functions in a structured graphical form.
  - **Identify and evaluate alternative system configurations.** The identification is a sequential process, starting from the identification of various combinations that can accomplish each task. The one is selected that best enables the subsystem to satisfy the performance criteria, given the constraints.
  - **Select the best configuration.** The analyst evaluates all subsystem configurations and adjusts the device mix so that conform to a single configuration.
  - **Design user interface, database and program.** The interface connects the user with the system and is thus extremely important. Databases and files are designed to store much of the data needed by decision makers in the organization. A well-organized database is the basis for all information systems.

Implementation Phase

- ✓ Implementation is the acquisition and integration of the physical and conceptual resources that produce a working system.
- ✓ The steps of the implementation phase are:
  - **Code and test system.** Programmers have a key role in this phase because they design, code and remove syntactical errors from computer programs. Then, testing of the program is carried out. A series of tests to pinpoint problems is run first with sample data and eventually with actual data from current system. [2]
  - **Install system.** Completed system that is fully tested can be installed for use.

Use Phase

- ✓ The use system consists of steps as below:
  - **Use the system.** Users use the system to meet the objectives that were identified in the planning phase.
  - **Audit the system.** After the new system has had a chance to settle down, a formal study is conducted to determine how well it is satisfying the performance criteria. Such a study is called a post implementation review.
  - **Maintain the system.** During the time the managers uses the system, modifications are made so that the system continues to provide the needed support. These modifications are called system maintenance, performed for three reasons:

- **To correct errors.** System use uncovers bugs in the programs or weaknesses in the design that were not detected in the system testing. These errors are corrected.
- **To keep systems current.** Over time, changes occur in the system's environment that requires modifications in the design or software.
- **To improve the system.** As managers use the systems, they see ways to make improvements. These suggestions are passed along to the information specialists, who modify the system accordingly. [1]

3.2 System Requirement

3.2.1 Requirement Eliciting

Before the requirements are captured, a few techniques have been used to find out what the users want. The main purpose is to assist in improving the current existing system and build more suitable functions for Human Resource Management System. The following are some techniques used to obtain user requirements:

- **Current situation research**

  Newspapers and journals and books were used extensively to find information on the current existing of human resource system. Studies are done on various issues, problems and solutions as written in the resources.

❖ **Face to face interviews**

I used this method of research to ascertain the majority of my information regarding the day-to-day operations and the type of work carried out by human resource department. The person I interviewed was Miss Iza Inora of LK Solutions, the Account and Administrative Executives.

❖ **Internet research**

The internet is the largest data and information warehouse. Most of the human resource management information can be found on the World Wide Web (WWW). I found a great deal of useful information on the WWW.

3.2.2 Functional Requirements

Functional requirements are a set of functions that are required to be included in the system. The functional requirements for HRMS are as the following:

a) Staff Information/Personnel Management

- Ability to view employees' own personnel data and allow them to manipulate their personnel data, including create and update their information.
- Allow administrators to view for any employee's personnel records in the system and view his or her information, including leave balance and training courses attended.

b) Leave Management

- Ability to user define all type of leaves (Annual, Meeting, Conference, Seminar, Study, Maternity, Compassionate, Unpaid, Paternity, Sick, Research, Vacation, Half Pay)
- Allows administrator to approve of the employees' leave application.
- Ability to cancel any pending or approved leave application.
- Tracking of Leave Balances (for various type of Leave) for each employee.

c) Claims and Benefit Management

- Ability to user define all type of benefits / claims (Car Allowance, Transport Allowance, Dental Allowance).
- Allow employees to apply for any acceptable claims and to be viewed by the administrator.
- Maintaining a list of panel doctors to administer medical claims

d) Training Courses

- The administrator can add new training courses that are available for employees.
- Ability to view list of current training courses available.
- Ability to modify and delete the existing lists of training courses.
- Allow employees to apply for the training courses offered.

e) Notice Management

- Allow all level of employees to post news or discussions that would like to be shared among employees.
- Ability to send, view and delete notice.

f) Search/Queries

- Allow users to search for certain employee to view his/her information, such as personal and work information, leave balance, total claims for current month and training courses attended.
- Allow administers to delete search results.

g) Security (for administrators only)

- The administrators can view employees' password to solve the problem of forgotten password.
- The administrators can view user log that is maintained of all access control activity, such as date and time of day and terminal identification.

3.2.3 Non-Functional Requirements

1) Reliability

A reliable system is a system that has minimum downtime and high information integrity. It does not produce dangerous or costly failures when it is used in a reasonable manner, which a typical user expects it as normal. Reliability is the extents to which a system can be expected to perform its extended function with required precision and accuracy.

2) Modularity

Modularity is a key factor in good program design. The working of the system was broken into modules so that distinct functions of objects could be isolated from one another. This characteristic makes testing and maintenance easier.

3) Usability

The system should be developed in such a way that is easy to use and user-friendly, so that users can interact with system comfortably and effectively. Visual effect and meaningful images and icons are used to provide the system with a sophisticated and yet simple to use user interface. Meaningful captions and menu options will simplify user communication with the system.

Confirmation message for any non-trivial process such as updating a record should be displayed. Effective error handling and validation procedures will also help the users to use the system. If an error occurs, such as invalid data input and invalid password, the system should display an error message to the user.

4) Simplicity

Simplicity refers to keeping forms and screens properly uncluttered in a manner that focuses the user attention.

5) Security

The system should be equipped with sufficient security. Each access should be authenticated by the system where users must login with the correct user ID and password to prevent unauthorized access into the system. The system should not show any potential of leakage of information. The password should be encrypted.

6) Efficiency

Efficiency is a process or a procedure that can be called or accessed in an unlimited number of times to produce similar outcomes or outputs at a creditable pace or speed.

7) Understandability

It is a degree of self-descriptiveness. The system should contain enough information for the users to determine its objectives, assumptions, constraint, input, outputs and status. No excessive information is presented. [3]

8) Maintainability

A product is maintainability if the system program are easily modified and tested in the case of updating a process to meet a new requirement, correcting errors or move to a different computer system.

3.3 System Analysis

3.3.1 Objectives of System Analysis

Objectives of the analysis are:

- ❖ To study the problem faced by the user
  - to identify the problem and find out the best solution to solve the problem.
- ❖ To study how human resource information is managed in the office environment
  - to study how the new system will improve the current items in reality.
- ❖ To acquire knowledge on how this system will be developed with the new emerging technology.
  - tools to develop the new system will be chosen among different types of new tools have been studied and stated in Chapter 2.
- ❖ To identify the major modules to be included in the system
  - to identify the modules that are feasible to develop and the knowledge and tools need to have in order to develop them.

3.3.2 Development Analysis

An analysis was carried out on the development tools to find out the most suitable tools for Human Resource Management System (HRMS). These tools include the entire platform, servers, development software and programming language. Besides considering the suitability of the tools to the requirements, the tools must be able to support and compatible with each other.

3.3.2.1 Server and Platform

Windows 2000 will be used as the HRMS platform since it is the upgrading of Windows 98 and enhancement of Windows NT 4.0. The system web server is Internet Information Services 4.0 (IIS), which is a built-in application in Windows 2000. Internet Explorer 5.0 is also included in Windows 2000 package, will be HRMS web browser.

Microsoft SQL Server 7.0 will be the system database to store data and information of human resources. SQL Server 7.0 is chosen in view of its scalability, high performance and tight integration with Windows 2000. Compared with Microsoft Access, SQL Server can handle larger amount of data during transaction and it is able to support more than 5 simultaneous users.

3.3.2.2 Development Software

To develop the system, Microsoft Visual Interdev 6.0 is used to build the application accessible from any platform running a standard Web browser such as Microsoft Internet Explorer. Visual Interdev has strong links with SQL Server, which makes it very easy to set up database combining ASP and SQL Server. It also allows to drag-and-drop pages from one location to another.

Adobe Photoshop 5.5 is used to create more interactive graphical user interfaces to produce a user-friendly and ease of use application. Meaningful images and icons are created using Adobe Photoshop.

3.3.2.3 Programming Language

I have decided to build HRMS using Active Server Pages (ASP) with JavaScript as its main scripting language (server-side scripting). The approach chosen is due to the reason that it is simple to implement and no additional software requirement needed besides Windows 2000. Abundant ActiveX controls are available for free in the Internet that ease development tasks. HTML is a default programming language in developing HRMS.

3.3.3 Conclusion

As a conclusion, this project will include the establishing and integration of the Leave Management System and other modules of Human Resource Management System. Following are the tools and technologies that used in this project:

Server and platform:

- Windows 2000 as platform
- Internet Information Services 5.0 (IIS) as web server
- SQL Server 7.0 as database
- Internet Explorer as browser

Development software:

- Microsoft Visual Interdev 6.0
- Adobe Photoshop 5.5

Programming language:

- Active Server Pages (ASP)
- ActiveX DLLs and Components
- HTML
- JavaScript

Other concept used:

- Structured Query Language (SQL)

References

[1] Raymond McLeod, Jr, Management Information Systems, Seventh Edition, Prentice Hall, Inc., 1998.

[2] Kendall & Kendall, System Analysis and Design, Fourth Edition, Prentice Hall Inc., 1999

[3] George M. Marakus, Decision Support Systems, Prentice Hall, Inc., 1999.

CHAPTER 4: SYSTEM DESIGN

Design maybe defined as the process of applying techniques and principles for the purpose of defining a device, a process or a system in sufficient details to permit its physical realization. It is a process in which requirements are translated into representation of software, transforming problem into solution. Initially the presentation gives an overview of the system. Subsequent refinement leads to a design representation that is very close to the source code.

The design of this system can be viewed from the following aspects:

- System functionality
- Database design
- User interface design

4.1 Overview of HRMS System

Figure 4.1 shows the overview of HRMS architecture to be built after the feasibility study and also the relationship between services and HRMS system.

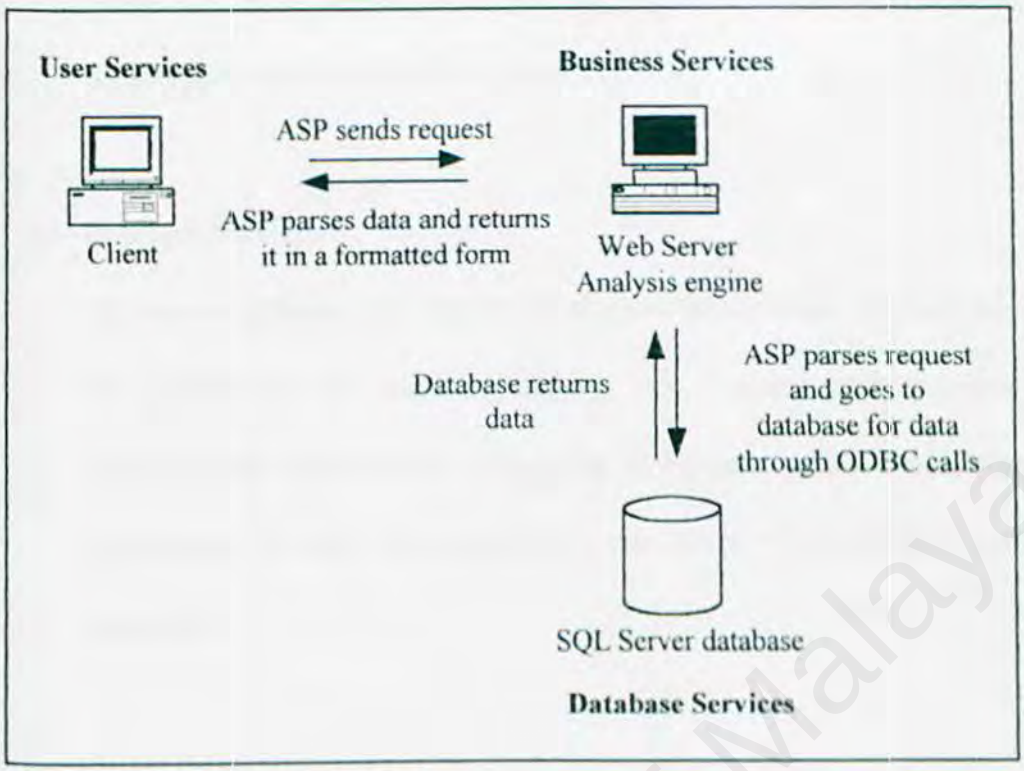

Figure 4.1 Overview of HRMS Architecture

HRMS is designed based on client-server architecture and extends it to the web. This architecture is divided into three distinct tiers, mainly three types of services – user services, business services and data services. Each of these services is used in the creation of system solution.

- **User services**

  Components in the user services tier provide the visual interface that a client will use to view information and data. Components in this layer are responsible for contacting and requesting services from other components in the user services tier or in the business services tier.

  In this level (which is the client browser), there is a component to gather input variables for analysis (for example, user request or query using

ActiveX Control hosted on a web page). There is also a component to display the result of analysis to the client.

- **Business services**

  At business services tier, there is an engine that performs the analysis. This tier resides on the machine naming IIS. Request and response are controlled by written codes specifying its business rules. These rules are coordinated by both client and server side script, which exists in an ASP document.

- **Data services**

  At the data services level, a repository of relevant data stored in the Microsoft SQL Server 7.0 database is available to support the work performed by the analysis engine.

4.2 System Functionality Design

System functionality design is based on the system requirements stated in Chapter 3. It translates the system requirements into system functionality. This design focuses on the system structured design and data flow design.

4.2.1 Data Flow Diagram (DFD)

A data flow diagram (DFD) is a graphic representation of a system that uses a small number of symbol shapes to illustrate how data flows through interconnected

processes. The data flow approach has four chief advantages over narrative explanation of the way data moves through the system. The advantages are:

- ✓ Freedom from committing to the technical implementation of the system too early.
- ✓ Further understanding of the interrelatedness of systems and subsystems.
- ✓ Communicating current system knowledge to users through data flow diagrams.
- ✓ Analysis of a proposed system to determine if the necessary data and processes have been defined. [1]

DFDs consist of only four symbols. The symbols are used to represent:

a) entity or environmental elements with which the system interfaces
b) processes
c) data flows
d) storage of data

Symbol	Name	Description
	Entity	It is used to depict an external entity that can send or receive data from the system.
	Flow of Data	It is to show movement of data from one point to another, with head of the arrow pointing towards the data's destination.
	Process	It is used to show the occurrence of a transforming process, input into output.
	Data Store	It is for showing data depository for data that allows addition and retrieval of data.

Table 4.1 Symbols Used in Data Flow Diagrams [2]

The following will illustrate the data flow diagram for each module in HRMS, including one diagram that depicts a general data flow of HRMS modules.

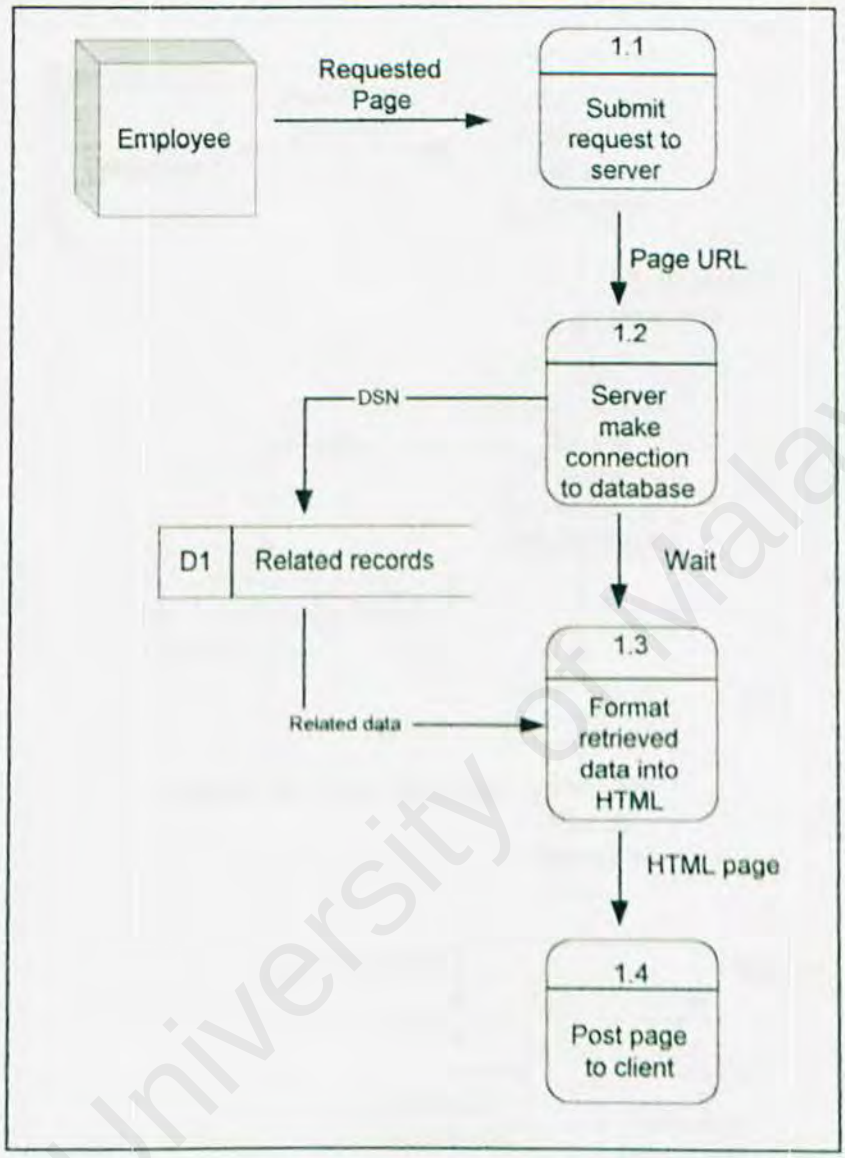

Figure 4.2 Data flow diagram for general data flow of HRMS modules

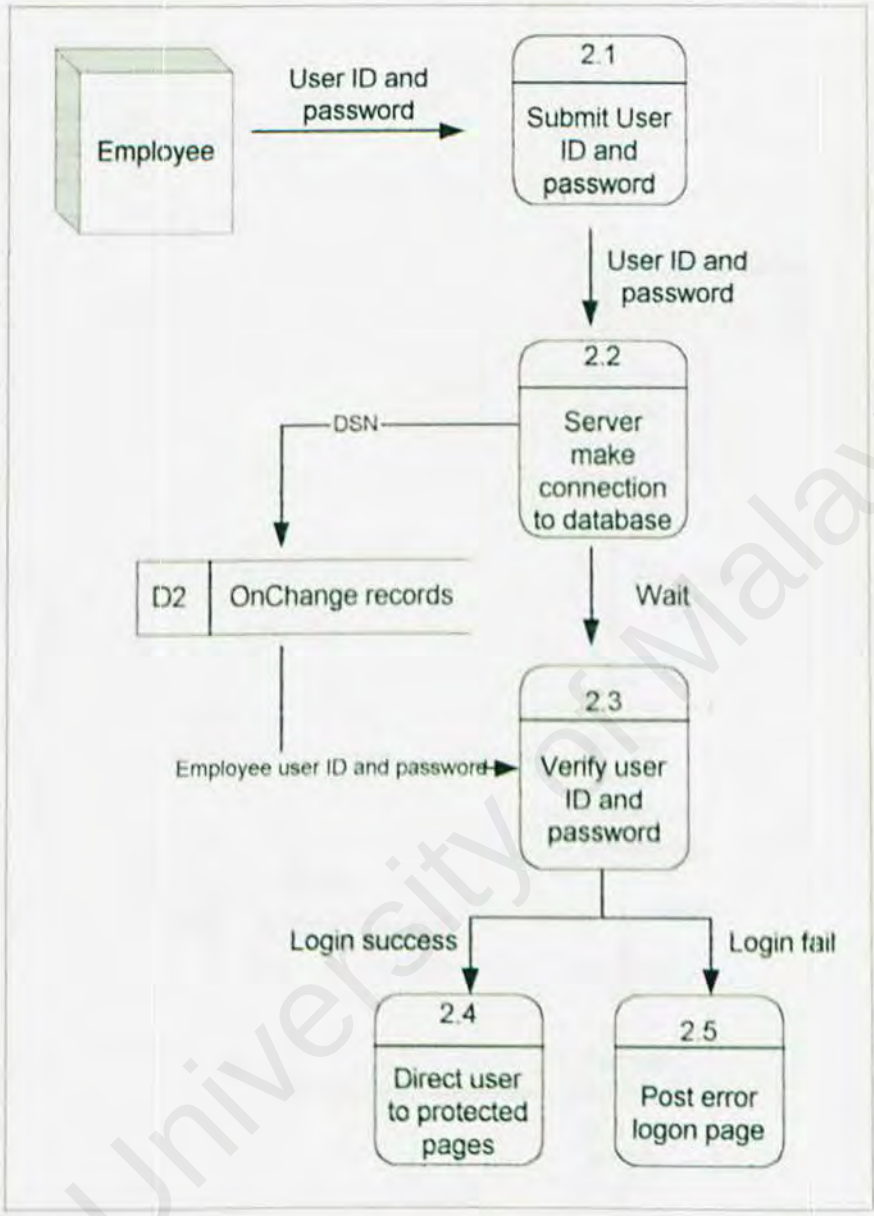

Figure 4.3 Data flow diagram for Authentication and Authorization modules

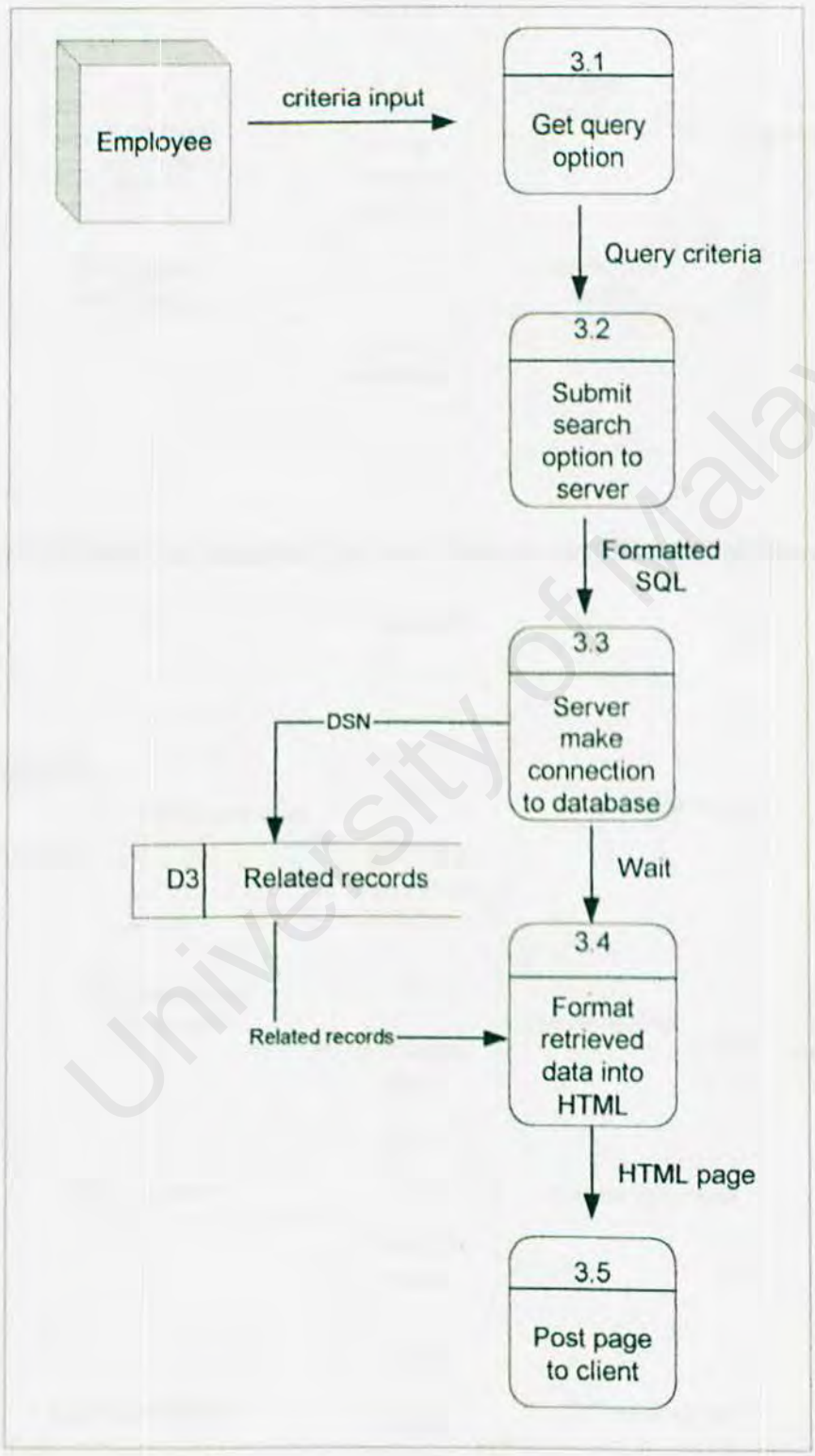

Figure 4.4 Data flow diagram for Search and Queries modules

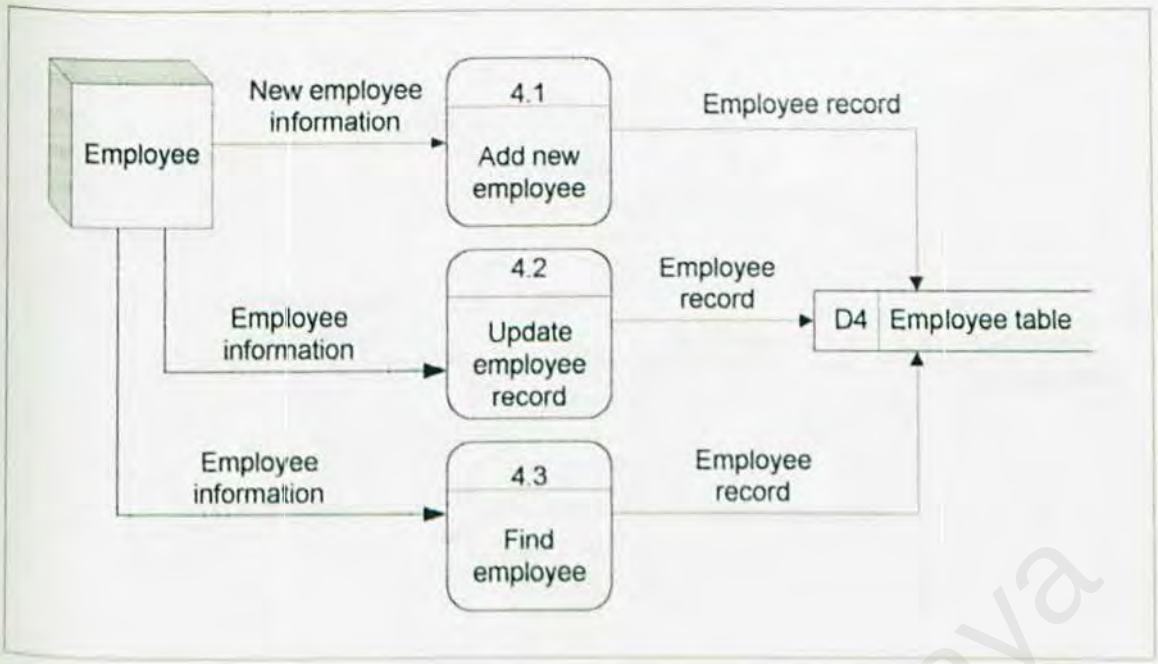

Figure 4.5 Data flow diagram for Staff Information/Personnel Management module

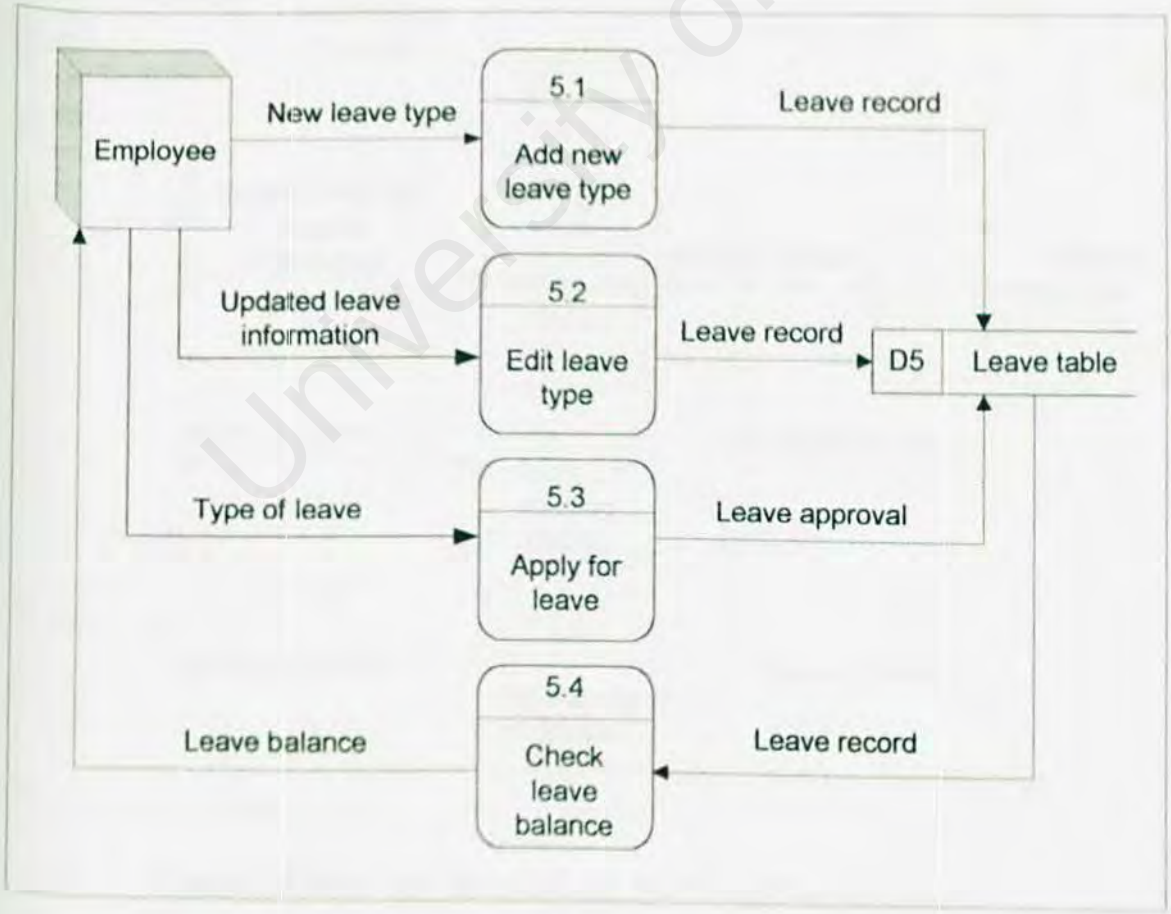

Figure 4.6 Data flow diagram for Leave Management module

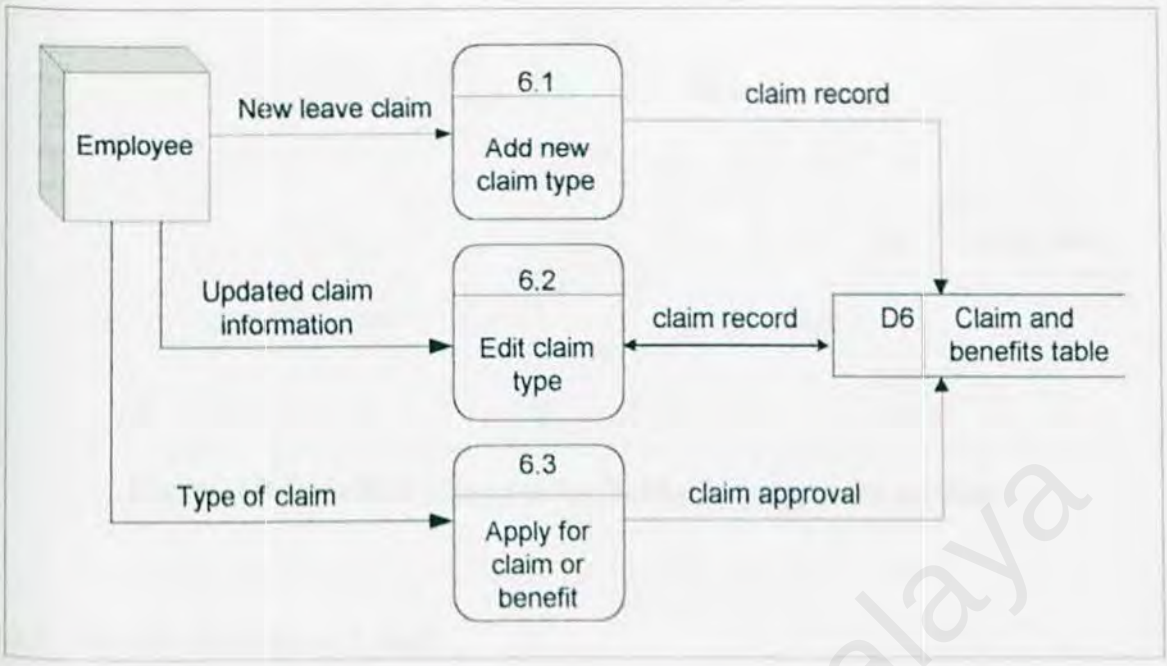

Figure 4.7 Data flow diagram for Claims and Benefits Management module

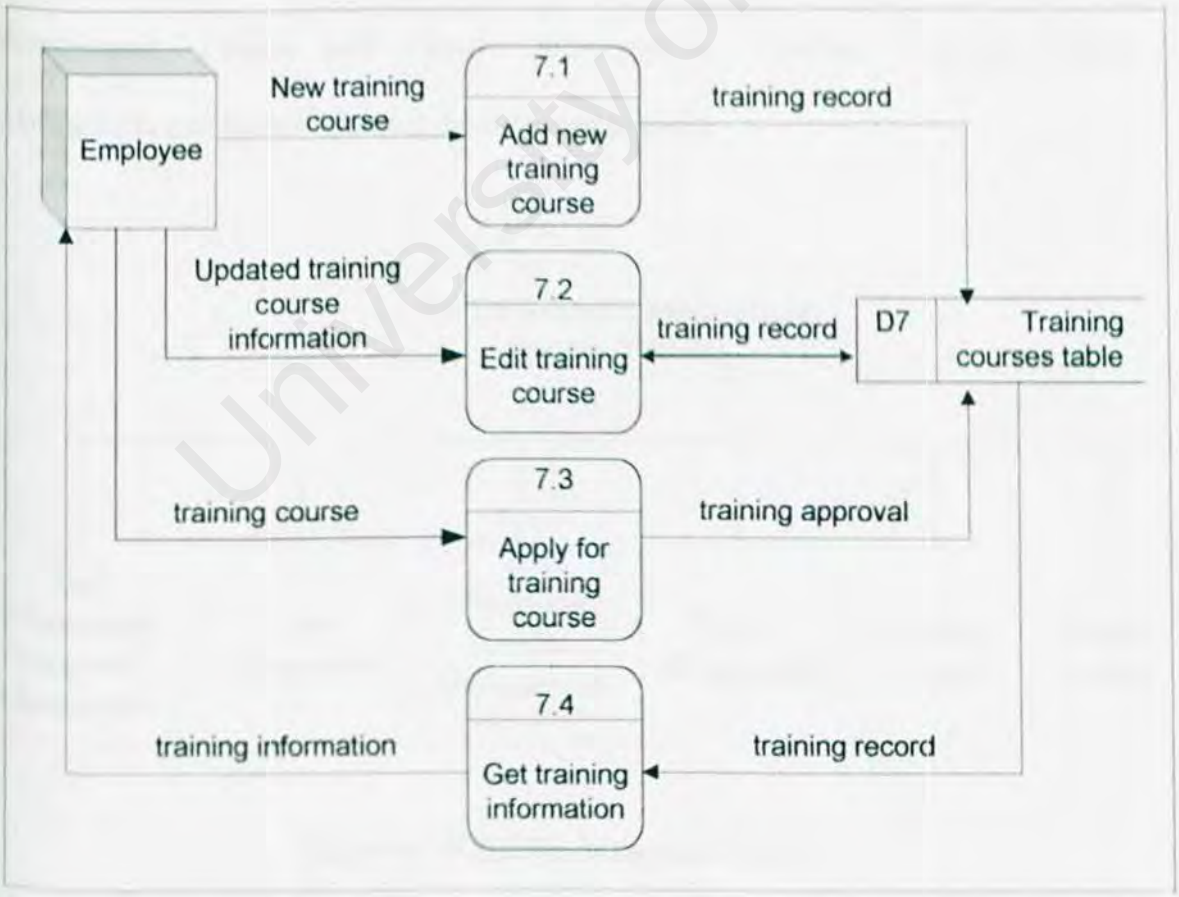

Figure 4.8 Data flow diagram for Training Courses module

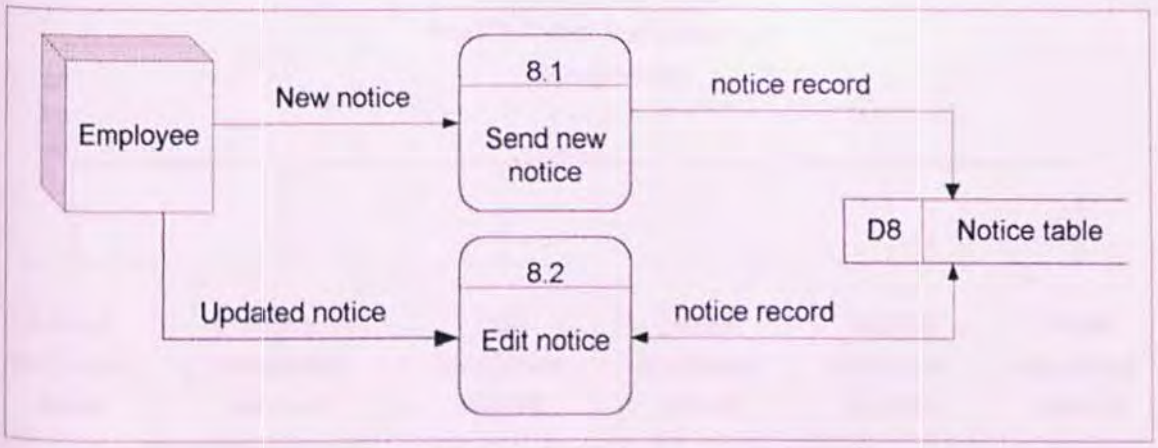

Figure 4.9 Data flow diagram for Notice Management module

4.2.2 System Structured Chart

The system structure is based in the functionality modules. The figures below show the system structure for Staff Information/Personnel Management, Leave Management, Claims and Benefit Management, Training Courses, Notice Management and Search/Queries modules respectively.

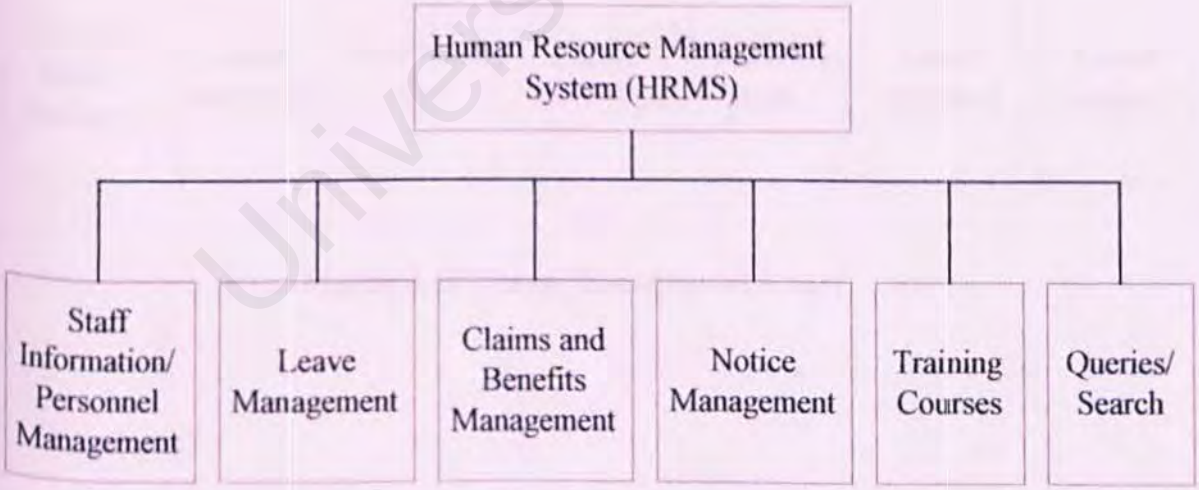

Figure 4.10 HRMS Structure Chart

Figure 4.11 Staff Information/Personnel Management Chart

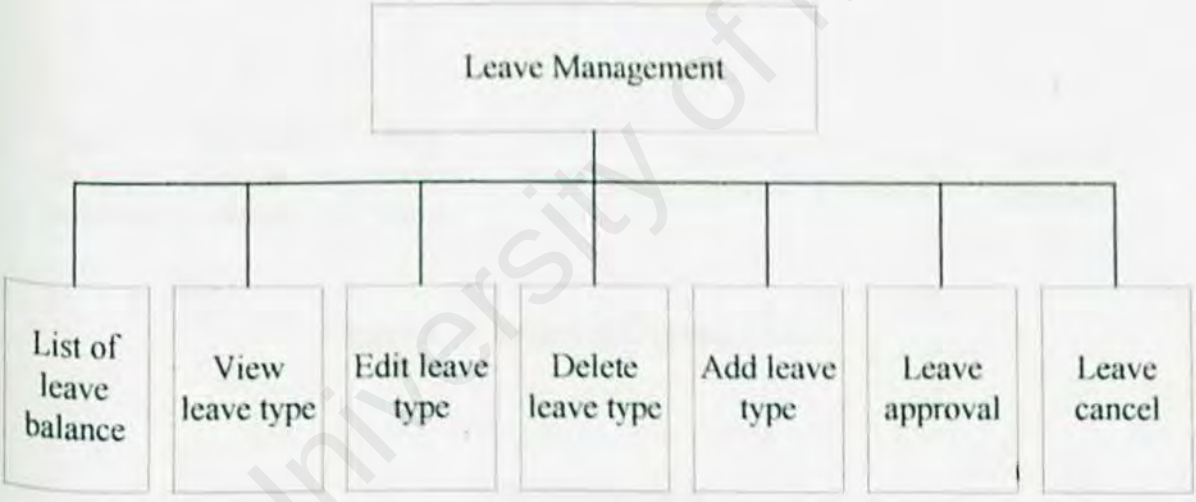

Figure 4.12 Leave Management Chart

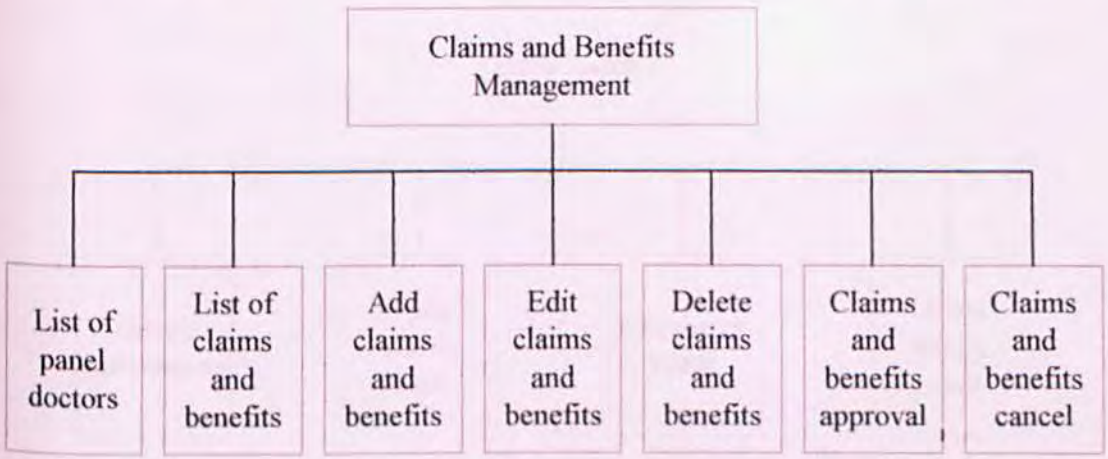

Figure 4.13 Claims and Benefits Management Chart

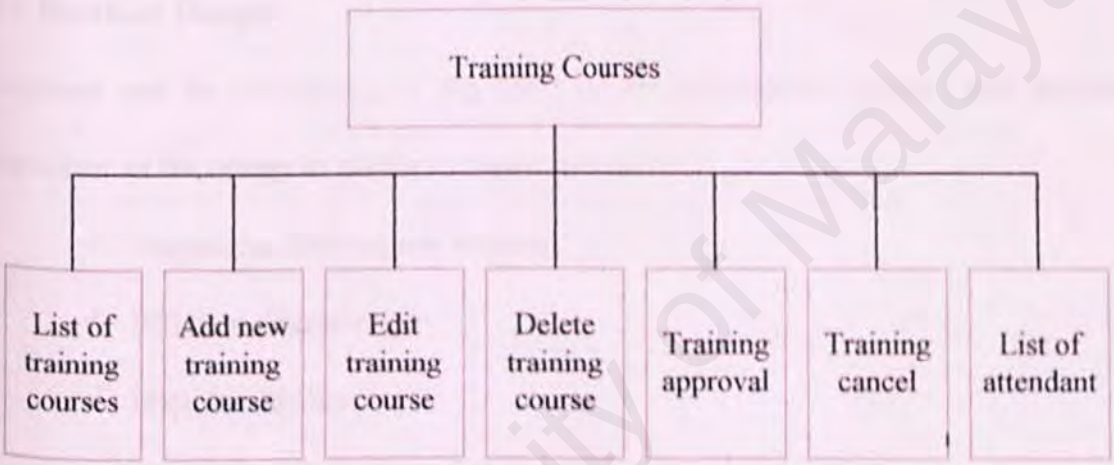

Figure 4.14 Training Courses Chart

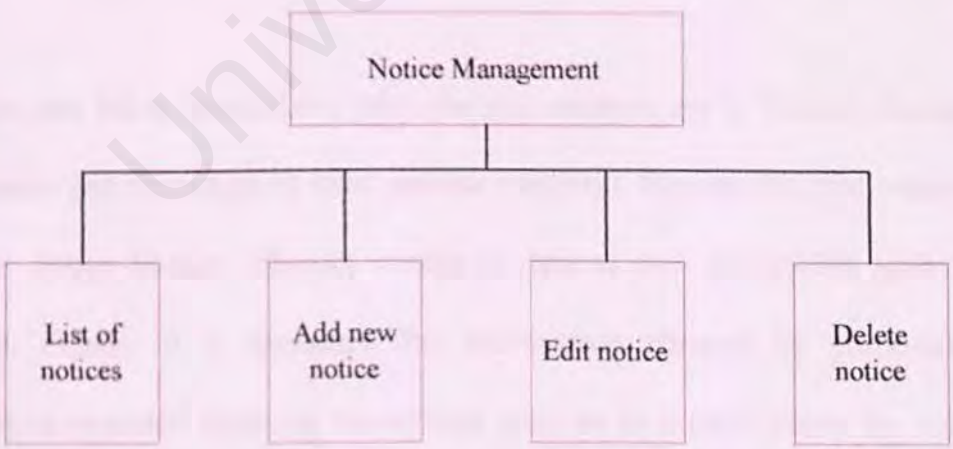

Figure 4.15 Notice Management Chart

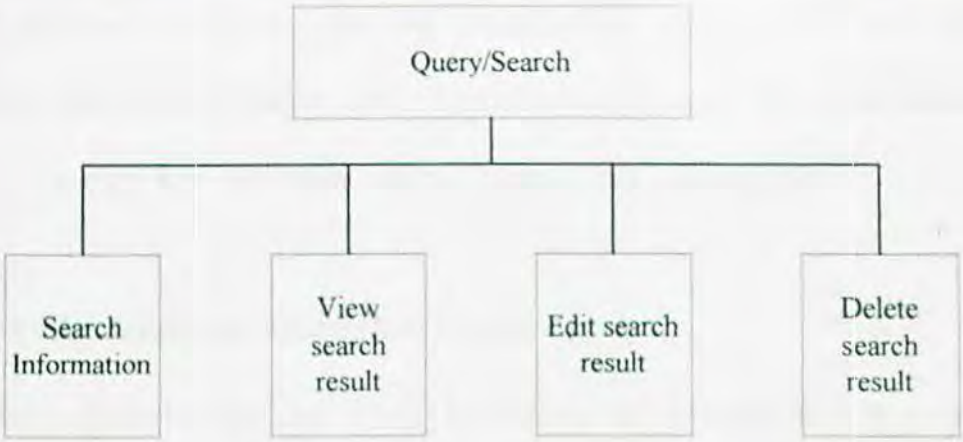

Figure 4.16 Query/Search Chart

4.3 Database Design

Database can be considered as the heart of an information system. The general objectives in the design of database organization are:

- ✓ Purposeful information retrieval
- ✓ Efficient data storage
- ✓ Data availability
- ✓ Efficient updating and retrieval
- ✓ Data integrity

First, the data has to be available when the user wants to use it. Second, the data must be accurate and consistent (it must possess integrity). Beyond this, the objectives of database design include efficient storage of data as well as efficient updating and retrieval. Finally, it is necessary that information retrieval be purposeful. The information obtained from the stored data must be in a form useful for managing, planning, controlling or decision-making. [2]

HRMS database is a relational database. In a relational database, a table is a collection of unique instances of similar data. Normalization reduces data redundancies and helps to eliminate data anomalies that result from those redundancies.

4.3.1 Entity-Relational Model (E-R Model)

An entity-relationship diagram (ERD) documents the system's data in a summary manner by identifying the types of data entities and their relationships. The ERD is a very flexible data-modeling tool, adaptable to various approaches that is used in system development.

The diagrammatic representation of HRMS database relationship can be illustrated in the entity-relationship diagram below.

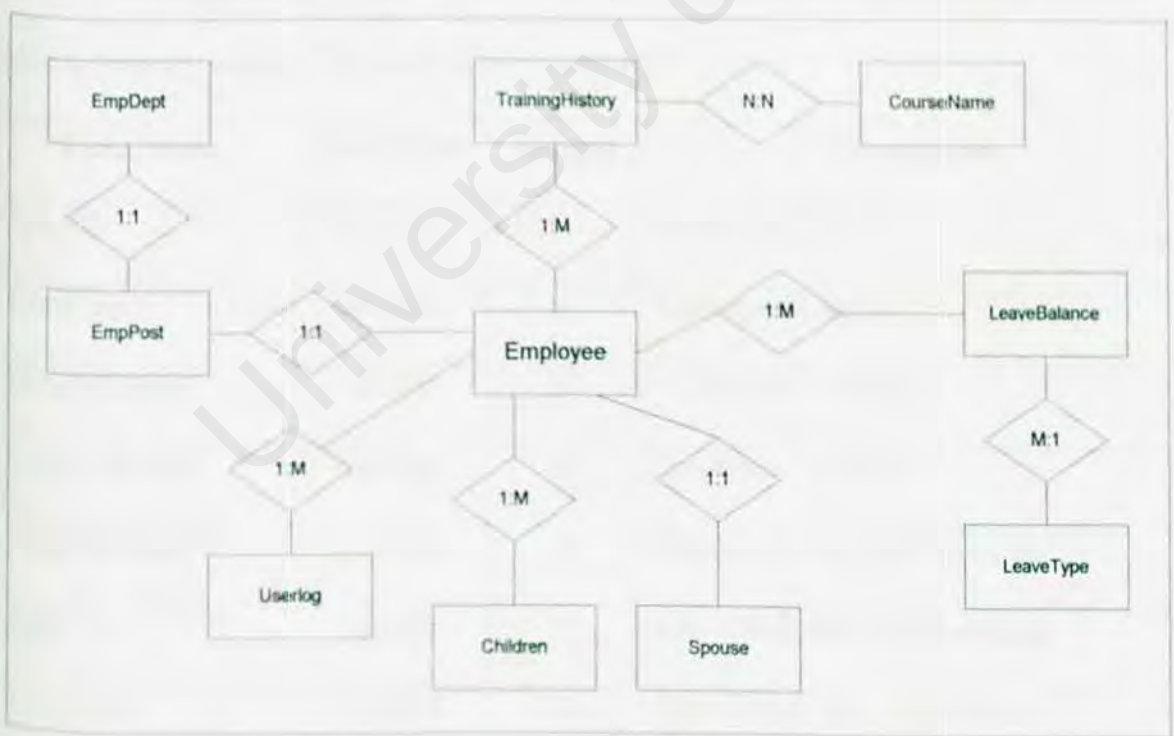

Figure 4.17 E-R Diagram for HRMS

4.3.2 Data Dictionary

A data dictionary is a user-accessible catalog of both database and application metadata. An active data dictionary is a dictionary whose contents are automatically updated by the Database Management System (DBMS) whenever changes are made in the database or application structure. A passive data dictionary is one whose contents must be updated manually when changes are made.

The database structure of relations in HRMS is listed as below. All relations are in third normal form.

A. Employee's personal table

This table consists of employee's personal record structure. The primary key for this table is EmpID, which is the employee's unique ID.

Field Name	Data Type	Length	Description
EmpID	varchar	10	Employee's ID
EmpName	varchar	50	Employee's name
EmpAddress1	varchar	50	Employee's address
EmpAddress2	varchar	50	Employee's address
EmpPostcode	varchar	5	Employee's address postcode
EmpCity	varchar	15	City where the employee stay
EmpState	varchar	25	State where the employee stay
EmpCountry	varchar	25	Country where the employee stay
EmpOldIc	varchar	10	Employee's old IC number

EmpNewIc	varchar	15	Employee's new IC number
EmpEmail	varchar	30	Employee's e-mail address
EmpTel	varchar	10	Employee's contact number
EmpSex	char	1	Employee's gender, male or female
EmpDOB	datetime	8	Employee's date of birth
EmpStatus	char	1	Employee's marital status, single or married
Nationality	char	2	Employee's nationality
Race	char	10	Employee's race
EmpPassword	varchar	10	Employee's login password
AnnualLeaveBal	int	4	Employee's annual leave balance
Medical LeaveBal	int	4	Employee's medical leave balance
LastLogin	datetime	8	Employee's last login to the system

Table 4.2 Employee's personal table

B. Employee's work related table

This table consists of employee's work related record structure. The primary key for this table is EmpID, which is the employee's unique ID.

Field Name	Data Type	Length	Description
EmpID	varchar	10	Employee's ID
EmpPost	varchar	30	Employee's position at work
EmpAdmin	char	1	Employee is admin/not admin
EmpDeptName	varchar	20	Employee's department name

EmpDeptCode	varchar	8	Employee's department code
EmpJoinDate	datetime	8	Employee's date of joining the company
EmpConfirmDate	datetime	8	Employee's confirm date of working in the company
EmpBasic	money	8	Employee's basic salary
EmpEPF	varchar	20	Employee's EPF number
EmpSocso	varchar	20	Employee's SOCSO number

Table 4.3 Employee's work related record table

C. Training Courses table

This table consists of training courses record structure. The primary key for this table is CourseCode , which is the course's unique ID.

Field Name	Data Type	Length	Description
CourseEmpID	varchar	10	Employee's ID
CourseName	varchar	100	Name of the training course
CourseCode	varchar	8	Code of the training course
CourseDate	datetime	8	Date of the course
CourseDuration	int	4	Days of the training
CourseVenue	varchar	50	Location of course taking place
CoursePresenter	varchar	30	Name of the course presenter
CourseFees	money	8	Fees of the course

Table 4.4 Training Courses table

D. Training Employee table

This table consists of list of employees that participate in certain training courses.

Field Name	Data Type	Length	Description
EmpID	varchar	10	Employee's ID
CourseCode	varchar	8	Code of the course that the employee apply
CourseApprove	char	1	Status of approval, Y or N

Table 4.5 Training Employee table

E. Leave Management table

This table consists of employee's leave record structure. The primary key for this table is LeaveCode, which is the leave's unique ID.

Field Name	Data Type	Length	Description
EmpID	varchar	10	Employee's ID
LeaveType	varchar	8	Type of leave type
LeaveApprove	char	1	Approval of applied leave
LeaveReason	varchar	500	Reason of taking leave
LeaveStartDate	datetime	8	Start date of leave
LeaveEndDate	datetime	8	End date of leave
LeaveDuration	int	4	Days of applying leave

Table 4.6 Leave Management table

F. Leave Type table

This table defines the types of leave and the entitlement for employees.

Field Name	Data Type	Length	Description
LeaveName	varchar	20	Type of the leave
Entitlement	int	4	Number of days entitled to employees for leave

Table 4.7 Leave Type table

G. Notice table

This table consists of notice management record structure.

Field Name	Data Type	Length	Description
NoticeSubject	varchar	30	Notice main title
NoticeDate	datetime	8	Date of the notice posted
NoticeContent	varchar	50	Content of the notice
NoticeSender	varchar	20	Name of the person who post the notice

Table 4.8 Notice table

H. Claims and Benefit table

This table consists of Claims and Benefits Management record structure.

Field Name	Data Type	Length	Description
EmpID	varchar	10	Employee's ID
ClaimMonth	int	4	Claim month
MileageDesc	varchar	150	Description of mileage claim
MileageAmount	money	8	Amount of mileage claim
TollDesc	int	150	Description of toll claim

TollAmount	money	8	Amount of toll claim
ParkingDesc	varchar	150	Description of parking claim
ParkingAmount	money	8	Amount of parking claim
OvertimeDesc	int	150	Description of overtime claim
OvertimeAmount	money	8	Amount of overtime claim

Table 4.9 Claims and Benefit table

I. Department Manager table

This table consists of list of manager for all departments in an organization.

Field Name	Data Type	Length	Description
DeptName	varchar	15	Department name
DeptManager	varchar	30	Department manager

Table 4.10 Department Manager table

4.4 Graphical User Interface Design

The interface is the system for all users. Well or poorly designed, it stands as the representation of the system. The objectives if designing user interface are:

- ✓ **Effectiveness** as achieved through the design of interfaces that allow users to access the system in a way that is congruent with their individual needs.
- ✓ **Efficiency** as demonstrated through interfaces that both increase the speed of data entry and reduce errors.
- ✓ **User consideration** as demonstrated in the design of suitable interfaces and by providing appropriate feedback to users from the system.
- ✓ **Productivity** as measured by ergonomically sound principles of design for user interfaces and workspaces.

The user interface has two main components: presentation language, which is the computer-to-human part of the transaction, and action language, which characterizes the human-to-computer portion. Together, both concepts cover the form and content of the term *user interface*. [2]

Some of the Human-Computer Interface (HCI) general principles of designing an interactive system have been considered and applied. These HCI general principles are listed in Table 4.11. [4]

Principles	Description
Consistency	Consistent format for command input, data display, button selection and placing of the control objects.
Confirmation and verification message	Ask for verification of any non-trivial destructive action such as record deleting.
Recoverability	Ability of the user to take corrective action once an error has been recognized.
Forgive mistake	The system should protect itself from user error that might cause it to fail.
Reverse action	Allow user to return to previous state (before change).
Function grouping	Categorize activities by function and organize screen geography accordingly.
Simple command name	Use short and meaningful command. Concise name is easy to memorize and reduce typing mistake.
Responsiveness	How the user perceives the rate of communication with the system. For example, the most pointer changes to hourglass or display a wait message when the system is processing data.

Table 4.12 HCI general principles

References

[1] Raymond McLeod, Jr, Management Information Systems, Seventh Edition, Prentice Hall, Inc., 1998.

[2] Kendall & Kendall, System Analysis and Design, Fourth Edition, Prentice Hall Inc., 1999.

[3] David M. Kroenke, Database Processing: Fundamentals, Design and Implementation, Sixth Edition, Prentice Hall Inc, 1999.

[4] R.S. Pressman, Software Engineering: A Practitioner's Approach, Fourth Edition, New York: McGraw-Hill Inc., 1997.

CHAPTER 5: SYSTEM IMPLEMENTATION

5.1 Introduction

The implementation phase of the system development is concerned with translating design specifications into real working version using programming languages. The primary goal is the production of a simple, clear source code with internal documentation that will ease the processes of verification, debugging, testing, modification and future maintenance. Source code clarity is enhanced by:

- structured coding technique
- good coding style
- appropriate supporting documents
- good internal comments
- meaningful names for variables being used
- indention of codes
- consistent convention in all the source code

5.2 Development Environment

The development environment consists of hardware and software requirement:

5.2.1 Hardware Requirements

The minimal hardware requirements for users are:

- At least Intel Pentium 233 MHz
- At least 64 MB RAM
- 512 K pipeline burst cache
- 24X CD-ROM drive

- 1.44 MB floppy drive
- 3.0 GB hard disk
- Other standard desktop PC components

5.2.2 Software Requirement

The following software specifications have been used to develop HRMS:

5.2.2.1 Software Tools for Development

During the development of HRMS, various software tools were used. Table 5.1 as shown below lists all the software used to develop HRMS.

Software	Module	Description
Microsoft Windows 2000	System requirement	Operating system
Internet Information Server 5.0 (IIS 5.0)	System requirement	Web server host
ODBC 32-bit Driver	System requirement	Connecting database with the web server
Microsoft SQL Server 7.0	Database	Database to store and manipulate the data
Microsoft Visual Interdev 6.0	System development	Coding the homepages
Internet Explorer 5.0	System development	Viewing the web pages developed
Microsoft FrontPage 2000	Interface design	ASP and HTML document layout design

5.2.3 Software Tools for Documentation

A software tool is any software product that can be used to facilitate the programming process and improve productivity. The most common software tool for documentation is the word processor. The *Microsoft Word 2000* is chosen because of the wide availability and of its user-friendliness. To draw the system model, chart and data flow diagram, *Visio 2000* is used because of its efficiency to create professional technical drawing.

5.4 Coding

The design must be translated into a form that can be 'understood' by the machine. The code generation step performs this task. HRMS is developed using ASP technology and Microsoft SQL Server 7.0 database application. The ASP enables a user to build web-based applications because it provides the Active Database Object (ADO), which provides easy access to any OLE/DB or ODBC compatible data source, including Microsoft SQL Server, as well as other popular databases from Oracle, Informix and Sybase.

5.4.1 Coding Approach

There are two approaches in coding, top-down approach and bottom-up approach.

Top-down approach allows the higher-level modules to be coded first before the lower level modules. The codes in the lower level module contain only and entry and an exit. This module is called a shell. The higher-level modules will reference the

lower level ones if they are coded and available. Reference to a shell will result in an empty action. This approach will ensure that the most important modules will be developed and tested first.

Bottom-up approach is the converse of top-down approach. It begins with the lower level modules first before the higher level modules. The higher-level modules are left as skeletons that call the lower level modules. This approach is used if the criticality of the lower level modules is high and need to be completed first. [1]

The method used for coding HRMS is *top-down method*. This approach is used to allow testing to begin with the main control module and works downwards and to detect unnoticed design errors at early stage.

5.4.2 Coding Style

Coding style is an important attribute of the source code and it determines the intelligibility of a program. An easy to read source code makes the system easier to maintained and enhanced. The elements of style include internal documentation (source code level), methods for data declaration and approach to statement construction.

5.4.3 Code Documentation

Code documentation begins with the selection of identifier names (variables labels), then the composition of commenting and ends with the organization of the program.

Use blank lines or indentation so that comments can be readily distinguished from code.

a) **Internal documentation**

Internal comments provide a clear understanding of source code during the maintenance phase of the software development. Comments provide the developer with a means of communicating with other readers of the source code. A statement indicating the function of the module and descriptive comments are embedded within the body of source code.

b) **Naming convention**

Naming convention provides easy identification for the programmer. It is created with the coding consistency and standardization.

c) **Modularity**

Modularity reduces complexity and facilitates change results in easier implementation by encouraging parallel development of different parts of a system. The various HRMS components are functionally independent from each other. New modules can be added and current modules can be modified without affecting other modules in the system.

5.4.4 Scripting Language

In HRMS, I have chosen to use JavaScript instead of VBScript. This is because JavaScript gives the ability to do things such as check form contents, communicate with the user based on their actions and modify the web page dynamically without the web page being re-loaded and without the use of Java, plug-ins or ActiveX controls.

Besides, JavaScript scripts run on the browser and are portable across any browser that includes JavaScript support, regardless of the operating system.

5.5 Input/Output

The style of the input and output follows the guidelines below:

- validate all input data
- display error message if user makes mistake
- display alert message of confirmation for certain tasks, such as deleting records of staff information
- keep the input format simple, uniform (standard) and user-friendly
- inform user if the task taken is successful or unsuccessful

References

[1] R.S. Pressman, Software Engineering: A Practitioner's Approach, 5th Edition, New York: McGraw-Hill Inc., 2001.

CHAPTER 6: SYSTEM TESTING

6.1 Introduction

Software testing is a critical element of system quality assurance and represents the ultimate review of specification, design and code generation. Once source code has been generated, software must be tested to uncover (and correct) as many errors as possible before delivery to the end user. [1]

During testing and debugging, I used the simulated data to perform all processing situations under specific conditions with the intent to find particular errors and then to debug them. In order to carry out a reliable and correct testing, test planning and strategies (will be discussed in Section 6.4) are derived.

6.2 Testing Objectives

Glen Myers states a number of rules that can serve well as testing objectives:

a) Testing is a process of executing a program with the intent of finding an error.

b) A good test case is one that has a high probability of finding an as-yet-undiscovered error.

c) A successful test is one that uncovers an as-yet-undiscovered error. [2]

In HRMS, the reasons and objectives for performing extensive tests during the design and development of the system are as follows:

- To achieve a high quality assurance such as completeness, accuracy, reliability and maintainability of the software program and its documentation.

- To ensure user satisfaction
- To ensure that the system can perform its function as expected
- To reduce cost in maintaining the system

6.3 Testing Process

Except for small programs, systems should not be tested as a single, monolithic unit. Large systems are built out of sub-systems, which are built out of modules, which are composed of procedures and functions. The testing process should therefore proceed in stages where testing is carried out incrementally in conjunction with system implementation.

Testing process in HRMS consists of five stages as shown in Figure 6.1.

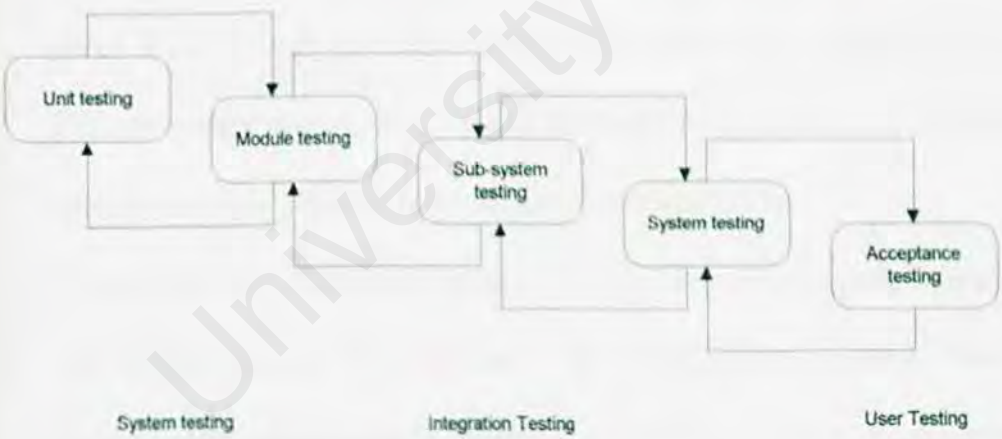

Figure 6.1 Testing Process

In general, the sequence of testing activities is component testing, integration testing then user testing. However, as defects are discovered at any one stage, they require

program modifications to correct them and this may require other stages in the testing process to be repeated.

The stages in the testing process as taken in HRMS are:

a) **Unit testing** Individual components are tested to ensure that they operate correctly. Each component is tested independently, without other system components.

b) **Module testing** A module is a collection of dependent components such as an object class, an abstract data type or some looser collection of procedures and function. A module encapsulates related components so can be tested without other system modules.

c) **Sub-system testing** This phase involves testing collections of modules, which have been integrated into sub-systems. The most common problems, which arise in large software systems, are sub-system interface mismatches. The sub-system test process should therefore concentrate on the detection of interface error by rigorously exercising these interfaces.

d) **System testing** The sub-system is integrated to make up the entire system. The testing process is concerned with finding errors, which result from unanticipated interactions between sub-systems and system components. It is also concerned with validating that the system meets its functional and non-functional requirements.

e) **Acceptance testing** This is the final stage in the testing process before the system is accepted for operational use. The system is tested with data supplied by the system procurer rather than simulated test data. Acceptance testing may reveal errors and omissions in the system requirements definition.

6.4 Testing Planning and Strategies

Test planning is concerned with setting out standards for the testing process than describing product tests. Test plans are not just management documents. [3] Test plan is a schedule and its objective is to design tests that systematically uncover different classes of errors and to do so with a minimum amount of time and effort. Throughout the development of HRMS, the following testing approaches have been practiced.

6.4.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. The unit test is white box-oriented, and the step can be conducted in parallel for multiple components.

The tests that occur as part of unit testing are illustrated schematically in Figure 6.2.

- The module *interface* is tested to ensure that information properly flows into and out of the program unit under test.
- The *local data structure* is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution.
- *Boundary conditions* are tested to ensure that the module operates properly at boundaries established to limit or restrict processing.
- All *independent paths* through the control structure are exercised to ensure that all statements in a module have been executed at least once.
- And finally, all *error-handling paths* are tested. [1]

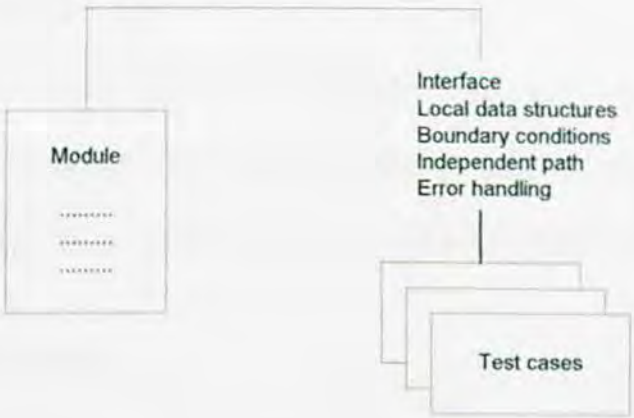

Figure 6.2 Unit Test

6.4.2 Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing – putting modules that have been unit tested together.

In HRMS, an incremental integration approach and top-down integration strategy is applied. **Incremental integration** is an approach where a program is constructed and tested in small increment. Errors are easier to isolate and correct; interfaces are more likely to be tested completely. **Top-down integration testing** is an incremental approach to construction of program structure. Modules are integrated by moving downwards through the control hierarchy, beginning with the main control module (main program).

6.4.3 System Testing

System testing is a series of different tests whose primary purpose is to fully exercise the whole system. Although each test has a different purpose, all work to verify that system elements have been properly integrated and perform allocated functions.

6.4.3.1 Security Testing

HRMS that manages sensitive information such as personnel data and staff particulars that can be improperly harm individuals is a target for improper or illegal penetration. Security testing attempts to verify that protection mechanism built into the system will, in fact, protect it from improper penetration. To quote Beizer [4]: "The system's security must, of course, be tested for invulnerability from frontal attack – but must also be tested for invulnerability from flank and rear attack."

6.4.3.2 Stress Testing

Stress tests are designed to confront programs with abnormal situations. For instance, test cases that require maximum memory or resources are executed. Stress testing executes system in a manner that demand resources in abnormal quantity, frequency or volume, particularly in handling the abnormal quantity of staff leave and training courses application.

6.4.3.3 Performance Testing

Performance testing is designed to test the run-time performance and software within the context of an integrated system. It occurs through all steps in the testing process. Even at the unit level, the performance of an individual module may be assessed as white box tests are conducted.

6.5 Test Case Design

6.5.1 White-box Testing

White-box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Using white-box testing methods, programmers are able to derive test cases that:

- guarantee that all independent paths within a module have been exercised at least once
- exercise all logical decision on their true and false sides
- execute all loops at their boundaries and within their operational bounds
- exercise internal data structures to ensure their validity.

White box testing is carried out at the early stages of the testing process. It is conducted to ensure that the internal operation of the system performs according to specification and all internal components have been adequately exercised.

6.5.2 Black-box Testing

Black-box testing focuses on the functional requirements of the software. That is, black box testing enables the programmer to derive sets of input conditions that will fully exercise all the functional requirements for HRMS. Black-box testing is not an alternative to white box techniques. Rather, it is a complementary approach that is likely to uncover different class of errors than white-box methods.

In HRMS, black-box testing attempts to find errors in the following categories:

a) incorrect or missing functions

b) interface errors

c) errors in data structures or external database access

d) performance errors

e) initialisation and termination errors

Unlike white box testing, which is performed early in the testing process, black box testing is applied during later stages of testing.

6.6 Conclusion

HRMS has been tested and debugged effectively to achieve the objectives of the system. Nevertheless, there is no foolproof testing method that will ensure that programs are free of errors. The best approach would be to use a combination of testing method – black-box and white-box testing together with inspection. To use all methods would, however, be very time consuming.

According to Roger S. Pressman [1], he mentioned that:

"Testing cannot show the absence of defects,

it can only show that system defects are present."

It is important to keep this statement in mind while testing is being conducted.

References

[1] R.S. Pressman, Software Engineering: A Practitioner's Approach, 5th Edition, New York: McGraw-Hill Inc., 2001.

[2] G.J. Myers, The Art of Software Testing, New York: John Wiley and Sons, 1979.

[3] Ian Sommerville, Software Engineering, 5th Edition. Edinburgh Gate: Addison Wesley Longman Limited, 1995.

[4] Beizer, B., Software System Testing and Quality Assurance, Van Nostrand Reinhold, 1984.

CHAPTER 7: SYSTEM EVALUATION AND CONCLUSION

7.1 Introduction

System evaluation is a process of evaluating the system that has been developed by identifying the problems encountered, system strengths and system constraints. It also highlights future enhancements of the system and the knowledge gained during development of the system.

7.2 Problems encountered and solutions

Throughout the process of developing HRMS, numerous problems have been encountered. The main problems and their solutions are listed below:

7.2.1 Lack of mastery in selected software

I have difficulty in writing and editing HRMS source code using Notepad. Therefore, I have chosen Microsoft Visual Interdev to assist me in writing the codes with ease. However, I was not exposed to this software before. To master it, I sought help and advice from friends and by reading related reference books. I also applied the knowledge gained from using Microsoft Visual Basic since it possesses the same characteristics as Microsoft Visual Interdev in writing codes.

7.2.2 Lack of knowledge in selected programming languages

Developing a web application is different compared to the stand-alone application. I used HTML, ASP and scriptings to develop this web-based system. I am lack of knowledge in writing and applying ASP codes to enhance my system. Searching for

articles and sample codes through the Internet and friends' assistance had solved this problem.

7.3 System strengths

7.3.1 User-friendly interface

HRMS is designed on the principle that it is ease of use. For example, graphical user interface (GUI) features have been integrated into the system to aid users of different levels. Users can easily capture the general idea of the system and hence, minimize the effort to train a new user.

7.3.2 Security feature

Every HRMS users are categorized into two types: the administrator and the non-administrator. Each user type is allocated certain access rights to the functions in HRMS. For instance, only the administrators are eligible to delete staff. Therefore, functions that are restricted will be either hidden or disabled. Thus, the possibility of an unauthorized access will be greatly reduced.

7.3.3 Authorization and authentication

HRMS only allows authorized users to access the system, which refers to users with correct login ID and password. An invalid login message will be displayed if a user tries to use the application without logging in or logging in with incorrect login ID or password. HRMS ensures its users to login before entering the system.

7.3.4 Informative message

HRMS provides error messages when a user attempts to perform illegal actions. Besides, it also provides messages after a certain task has been completed. The main

purpose of showing messages is to let the users to understand what is going on and to keep them informed of what has been done.

7.3.5 Consistency

The screen design is consistent throughout the whole system. The menus are always displayed at the same position although the user switches from one module to another. Users can easily seek for a particular option that they require in the system.

7.3.6 System transparency

System transparency refers to the condition where the users do not need to know where the database resides, the structure of the system, the database management system (DBMS) or any issue related to the architecture of the system.

7.4 System constraints

Due to the time constraint, some of the system features could not be implemented. A number of limitations of the system are as outlined below:

7.4.1 Database not encrypted

Employees information stored in server database is not in secured condition because it is stored in a plain text rather than in encrypted format.

7.4.2 Insufficient functionality

HRMS that was created does not reflect a real HRMS for an organization. This system is only a miniature of a complete HRMS.

7.4.3 One level approval of user's application

Training and leave applications are approved by one level of management. Once an authorized person (manager, head of department or officer) has approved it, the user's application is considered approved by other higher levels of management.

7.5 Future enhancement

There are some features that can be added into the current system for enhancement.

7.5.1 Provide report generation

In the real world of HRMS, various reports are generated electronically for the purpose of printing them out to be viewed by the administrators, such as list of employees that take leave and the list of employees that join a certain training course. Such reports are no longer being prepared manually. Therefore, it should be considered for future enhancement.

7.5.2 Provide ability to send e-mail

After the administrator has approved a certain employee's application, such as leave application, notification of leave approval is preferably sent by e-mail. This is because people usually check their mails very frequently, almost everyday.

7.5.3 Provide more system functionalities

HRMS should add more functions, such as Payroll System, Attendance System and Recruitment Management. Adding these features is more likely to build a complete Human Resource Management System.

7.6 Knowledge and experience gained

Throughout the development of HRMS, I have gained a lot of knowledge especially in mastering the web development tools, such as Microsoft Visual Interdev, SQL Server and ASP programming. Although I was not exposed to these tools before, I am glad that I am able to learn and use them in a short time.

Besides that, the knowledge gained from this project is the awareness of user's needs. It was found that users need a user-friendly environment, a readable homepage and clear instructions and guidance. And the programmer needs to find and organize the information acquired and analyze the behavior

One important thing I realized is that it is really useful and important to have an up-to-date knowledge and information in keeping up with the fast and ever changing technology of information technology. This project has proved very beneficial for me when I step into the working environment in future.

Conclusion

This project has been a valuable one because the knowledge gathered throughout the three years course in university has been applied, particularly on the use of software engineering, system analysis and design and database management.

At the completion of this project, HRMS has achieved its objectives, as well as the functional and non-functional requirements as planned at the beginning of the project. It will serve as a web-based application for all types of organizations. HRMS is believed to become more powerful as the suggested enhancements and features are added, in near future.

USER MANUAL

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

HRMS is a web-based application that helps employees to manage their own tasks, such as applying for leave, training courses and claims. It also enables the employees to have discussions that they would like to share with other staffs.

The main advantage of HRMS is that it is not a stand-alone application. Employees can access it anywhere as long as they are connected to the Internet.

ABOUT THE USER MANUAL

HRMS consists of employee-based module and administrator-based module. The difference between the two modules is that employee-based module (Part 2) is designed for all employees including administrators and non-administrators. Certain functions are restricted to non-administrators and can only be used by HRMS administrators. These functions are put in administrator-based module as in Part 3.

Briefly, HRMS user manual consists of four parts as the following:

- **Part 1: Getting Started with HRMS**
  - Hardware and software requirements
- **Part 2: Employees Mode**
  - Login to HRMS
  - Registration for new user
  - Leave Application
  - Claims and Benefit Application
  - Training Courses Application
  - Inbox Management

- Update Registration Information via Change Password

- **Part 3: Administrator Mode**

  - Search employee information
  - Edit the training courses information
  - Add new training courses
  - Approve employees' applications
  - Delete overdue items and old discussions

- **Part 4: Quitting HRMS**

  - Logout

PART 1: GETTING STARTED WITH HRMS

HRMS is a web-based application that does not require installation. Typing the address in the web browser can access HRMS. Before accessing HRMS web site, make sure that your computer meets the requirements as stated in the following:

1.1 Hardware and software requirements for server

The minimal hardware requirements are:

- At least Intel Pentium 233 MHz
- At least 64 MB RAM
- 24X CD-ROM drive
- 1.44 MB floppy drive
- 3.0 GB hard disk
- Modem (minimum 14.4 kbps) or network connection through existing network configuration
- Other standard desktop PC components

The software requirements are:

- Microsoft Windows 2000
- Internet Information Server 5.0 (IIS 5.0)
- ODBC 32-bit Driver
- Microsoft SQL Server 7.0

1.2 Hardware and software requirements for client

The minimal hardware requirements for users are:

- At least Intel Pentium 233 MHz
- At least 64 MB RAM

- 24X CD-ROM drive
- 1.44 MB floppy drive
- 3.0 GB hard disk
- Modem (minimum 14.4 kbps) or network connection through existing network configuration
- Other standard desktop PC components

The software requirements are:

- Internet Explorer 5.0 or above as browser

After the above requirements have been fulfilled, HRMS can be used. You may start opening your browser and type HRMS web site address on the browser *Address* box and press *Enter* to access it. If the server is available, the first page of HRMS login page will appear. Figure 1.1 shows the HRMS homepage.

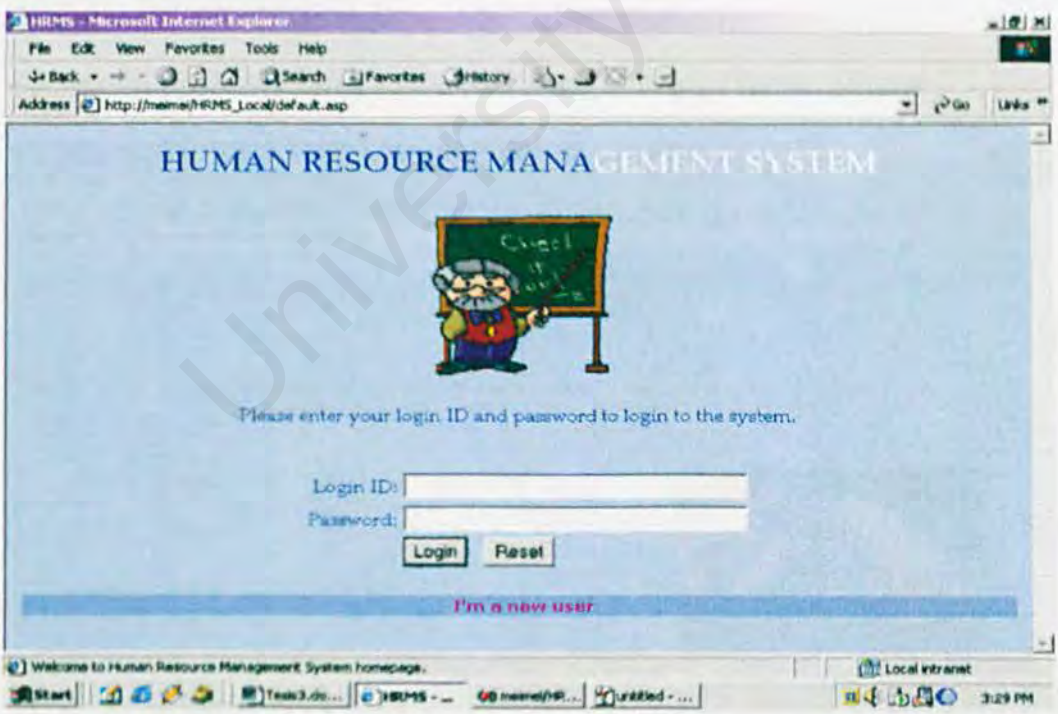

Figure 1.1 HRMS homepage

PART 2: ALL EMPLOYEES MODULE

This module is created for all employees, including the administrators and the non-administrators.

2.1 Login to HRMS

1) From the homepage, login is provided for registered users, as shown below.

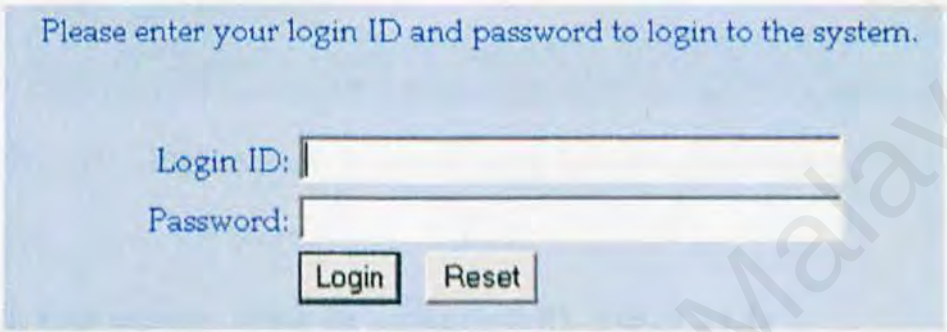

Figure 2.1.1 Login

2) Users are required to type in their login ID and password correctly in order to enter the system successfully.

3) Successful login will bring to the next page, which is 'Home' (Figure 2.1.2).

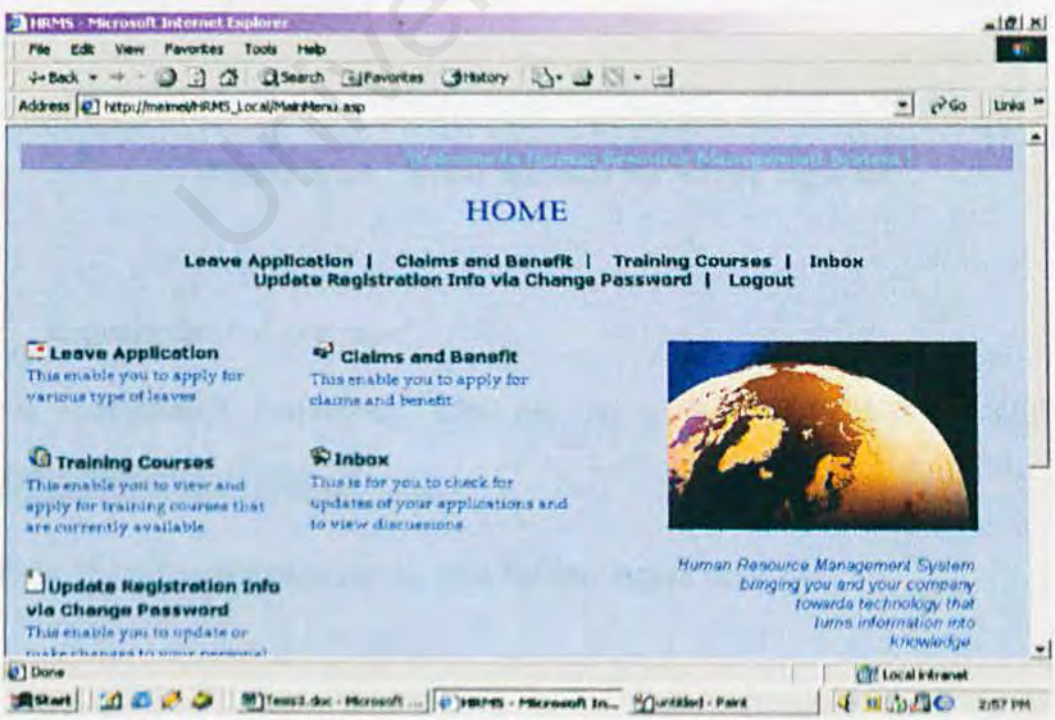

Figure 2.1.2 Home

4) Unsuccessful login will bring to this error message for wrong password entered, as in Figure 2.1.3,

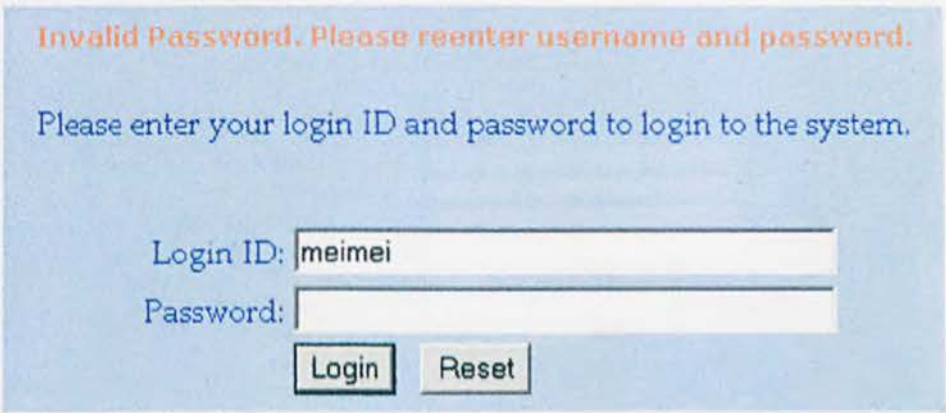

Figure 2.1.3 Error message for wrong password

or this error message below for wrong login ID. (Figure 2.1.4)

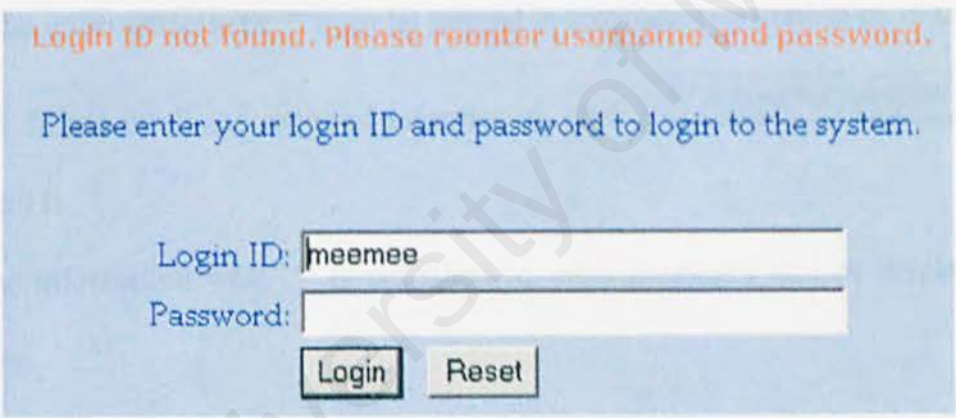

Figure 2.1.4 Error message for wrong login ID

2.2 Registration for new user

1. For unregistered employees, they can be a user of HRMS by clicking I'm a new user to register.

2. Then, the following page can be seen for user registration.

Figure 2.2.1 New user registration form

3. All the information with '*'must be entered to complete registration successfully.
4. After filling in all the required information, click Submit Registration to submit it.
5. If the information with '*' is not filled in, error messages will be displayed as below:

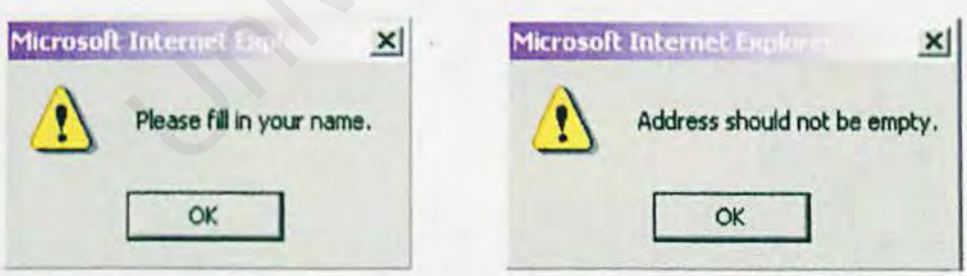

Figure 2.2.2 Error messages for user registration

6. Successful registration will lead you to the message as shown in Figure 2.2.3.

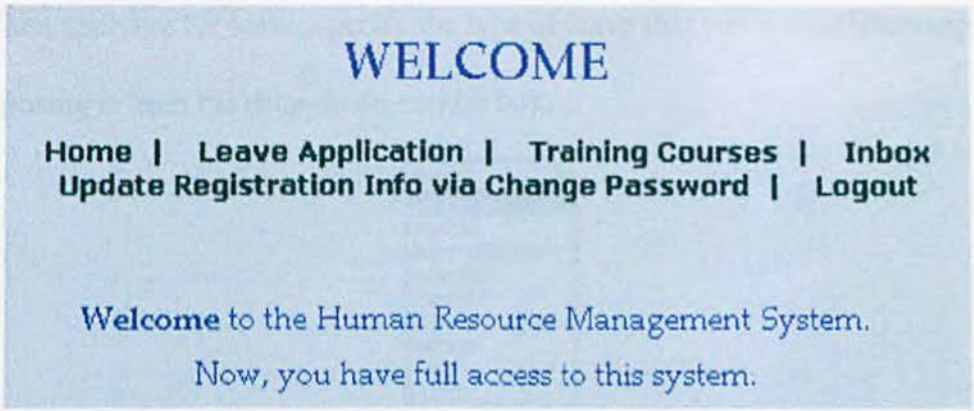

Figure 2.2.3 Welcome page for successful registration

2.3 Leave Application

This module is created for employees to apply for leave electronically without having to fill in the application form manually on the paper. To apply for leave:

1) Click **Leave Application** to go to the page shown in Figure 2.3.1.

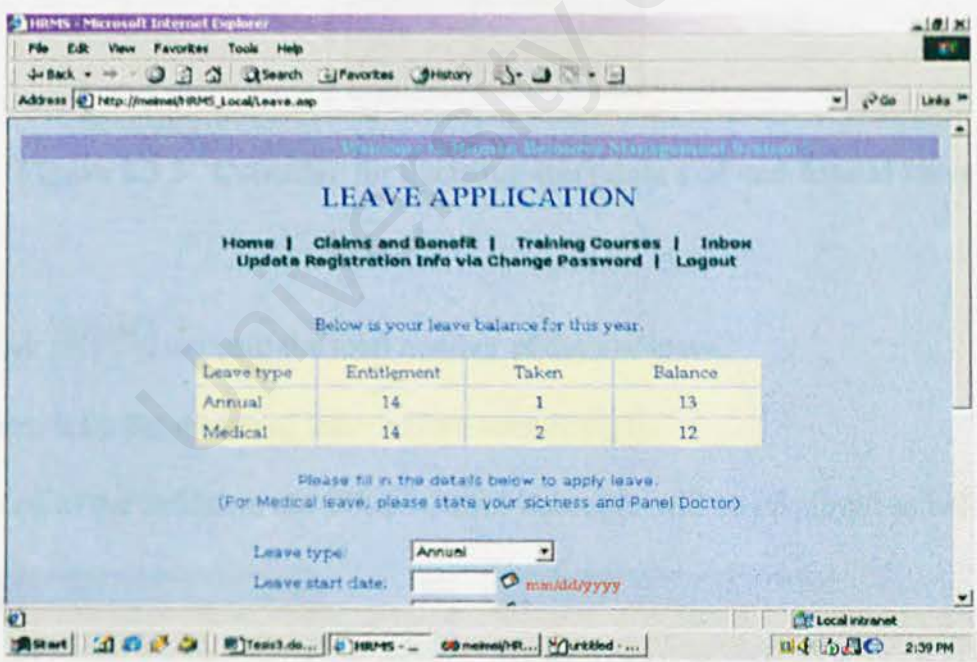

Figure 2.3.1 Leave application form

2) From the Leave Application form, leave balance for the current year is shown.

3) When applying for leave, specify the type of leave that you would like to apply by choosing it from the drop-down combo box.

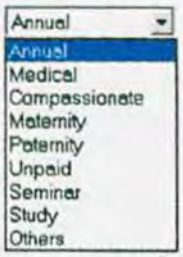

Figure 2.3.2 Leave type

4) Then, fill in the start date and end date of leave by clicking to view a calendar. Click the day on the calendar to choose the day of leave.

Figure 2.3.3 Calendar for choosing start date and end date of leave

5) Click Count to count the total number of days of leave.

6) Then, state the reason of leave before submitting it.

7) If one of the fields are not filled in, error messages will be displayed as below,

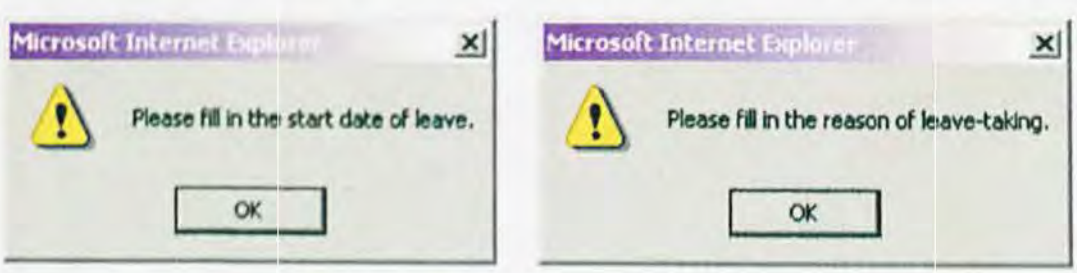

Figure 2.3.4 Error messages for leave application

else the leave application will be submitted successfully, as shown in Figure 2.3.5.

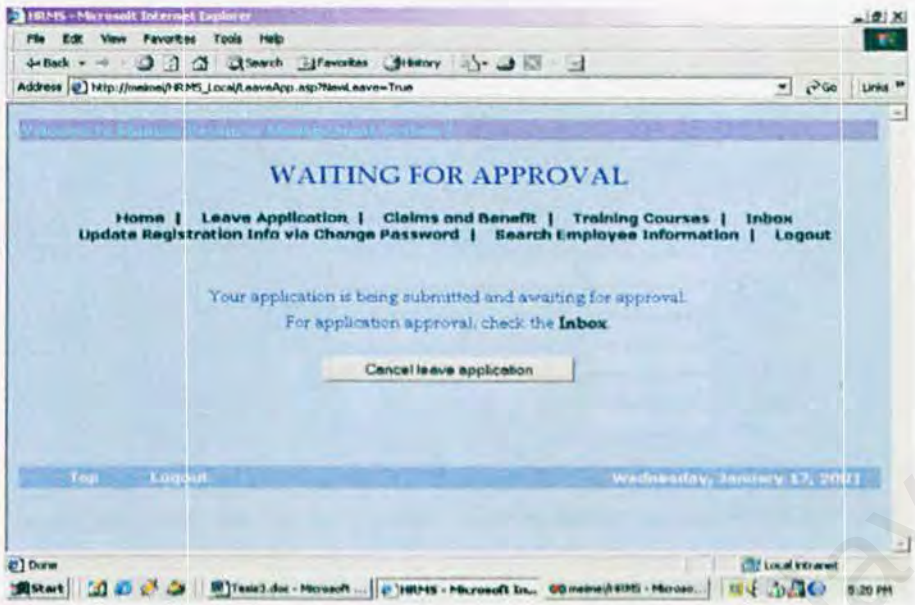

Figure 2.3.5 Leave application is submitted

8) To cancel the leave that has just applied immediately, you can click Cancel leave application or cancel it in your "Inbox".

2.4 Claims and Benefit Application

Claims and benefit application is for employees to make claims for current month. Employees can update their claims daily for that particular month.

1) To apply for claims and benefit, click **Claims and Benefit** to go to the following page for application form.

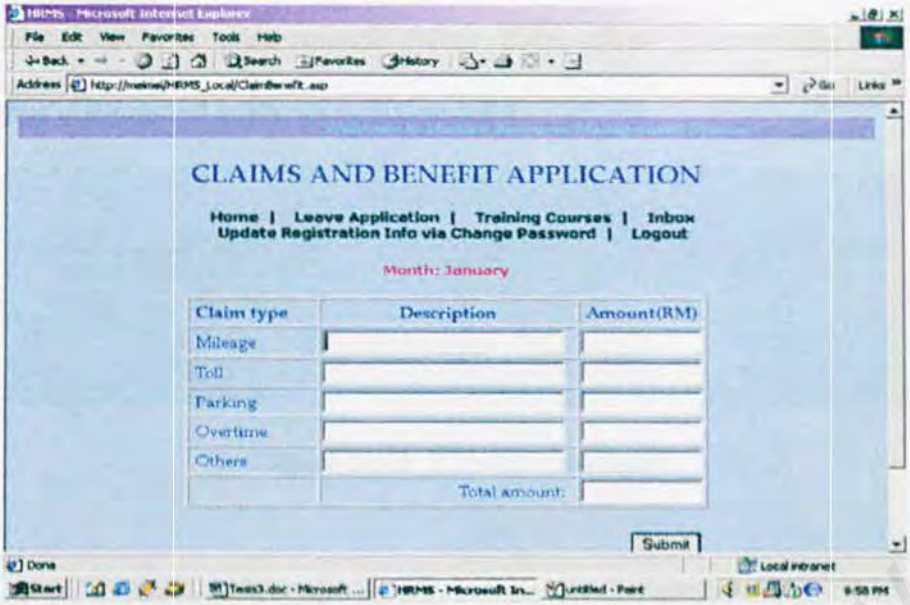

Figure 2.4.1 Claims and benefit application form

2) From Figure 2.4.1, users can claim for various types of claims as above and state the amount of money to claim.

3) If the form is already filled in, you just need to update it by adding appropriate descriptions and total amount of claims. The sample for filled form is:

Month: January

Claim type	Description	Amount(RM)
Mileage	Batu Tiga, Shah Alam	5
Toll	LDP toll	15
Parking	PJ area	1.2
Overtime	2/1/2001	60
Others		0
	Total amount:	81.2

Figure 2.4.2 Claims form that is filled in and needs to be updated

5) Click Submit to update monthly claims. Users will be informed after the claims is submitted as shown in Figure 2.4.3.

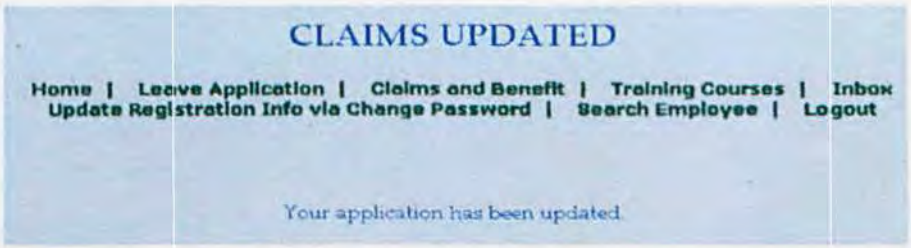

Figure 2.4.3 Claims submitted

2.5 Training Courses Application

Courses that are available can be applied by all employees and then are waiting for approval by the department manager. Approved application will be notified by checking it at "Inbox".

1) To apply for a course, click Training Courses to view the list of courses available.

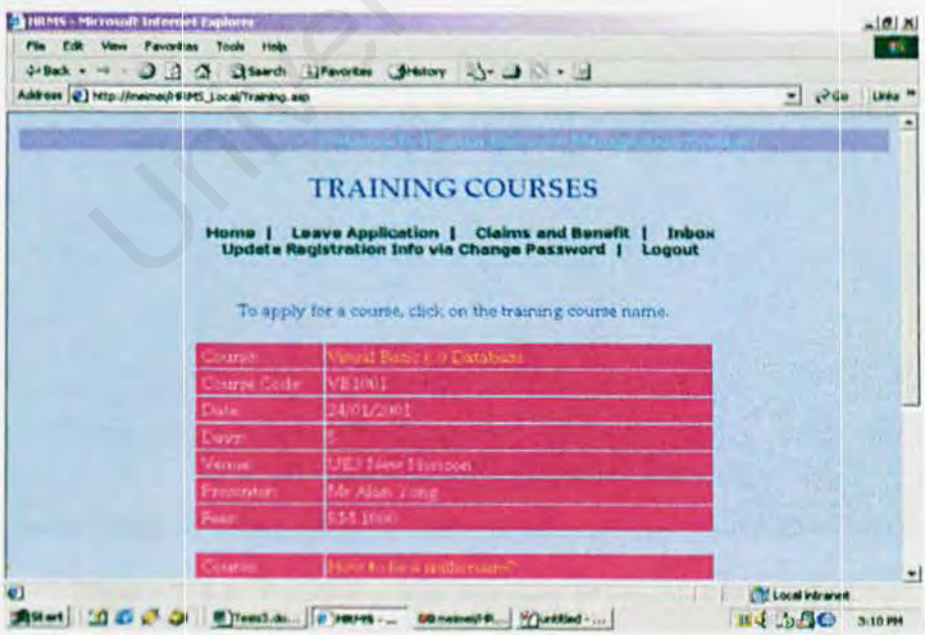

Figure 2.5.1 List of training courses available

2) After viewing the list and you decide to apply for a certain course, click on the course name.

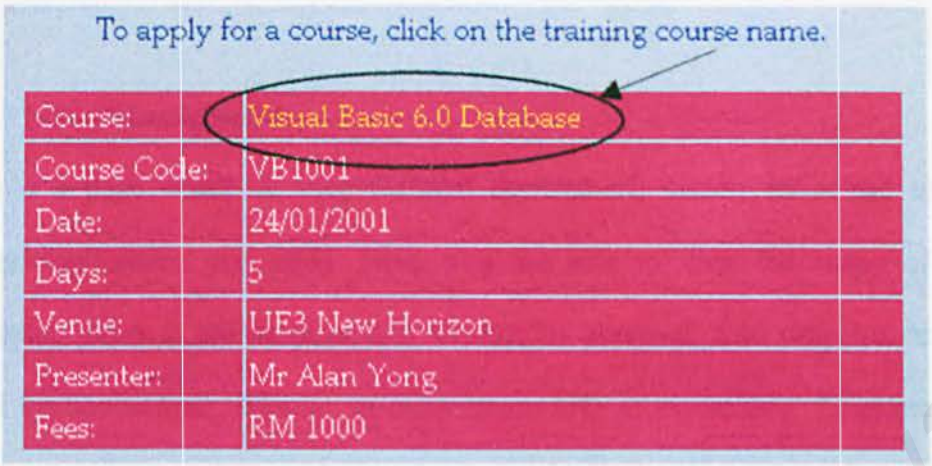

Figure 2.5.2 Click on course name to apply for a course

3) After clicking, a message of confirmation about applying the course is showed. If you are sure to apply it, click Confirm and your application will be submitted, waiting for approval.

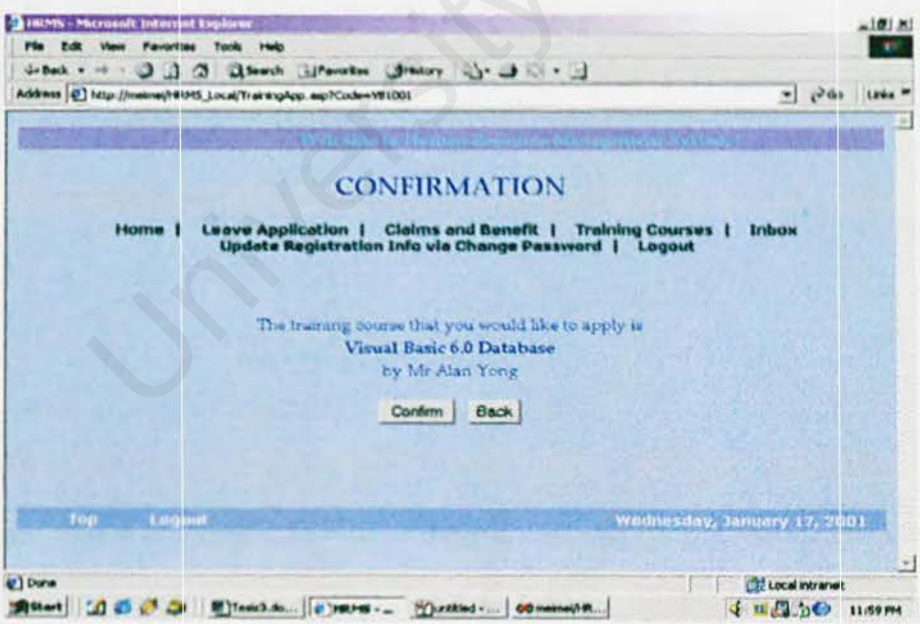

Figure 2.5.3 Confirmation message of course application

4) To cancel the course that has just applied immediately, you can click

Cancel training application or cancel it in your "Inbox".

2.6 Inbox Management

Inbox is a page where all your personal applications, namely leave and training courses applications are listed. Here, you are able to view the status of your application either is approved or still waiting for approval. You can also edit and cancel the applications. Besides, discussions are held in this section. Employees can view current discussions and post a new discussion.

1) To view "Inbox", click **Inbox**, as seen in Figure 2.6.1. The first portion of "Inbox" is the "Personal Item" and the second portion is about "Discussions".

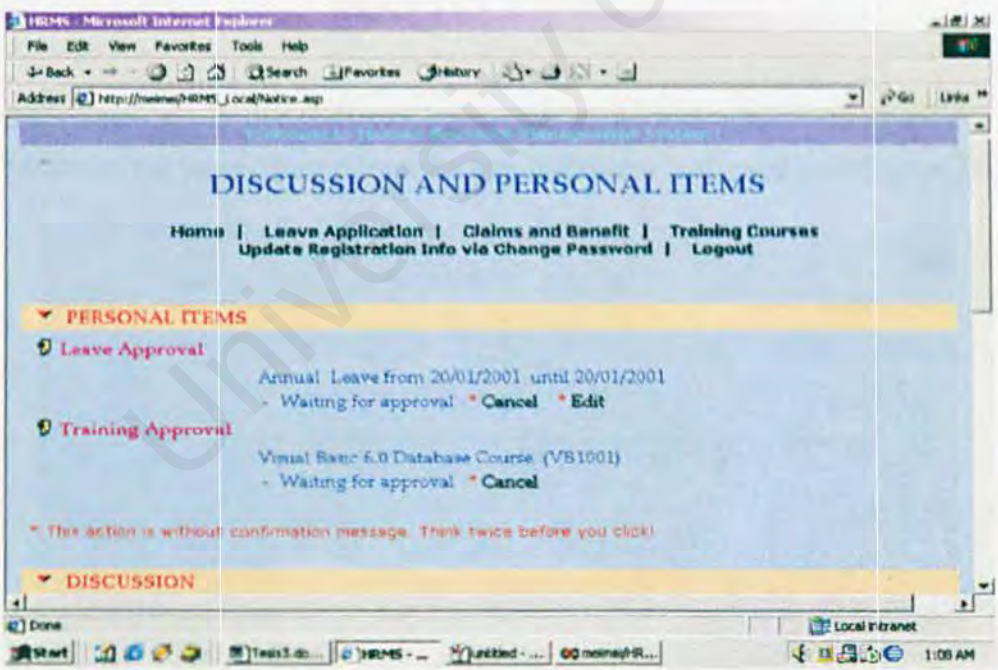

Figure 2.6.1 Inbox

2) In "Personal Items" portion, users can edit the leave applied previously by clicking * Edit or cancel the leave or training courses applied by clicking * Cancel. The '*' warns users that all actions taken, including editing and deleting is without confirmation message. No "Undo" function is provided for actions taken.

3) In "Discussion" portion, users are able to view all discussions posted by employees as shown below:

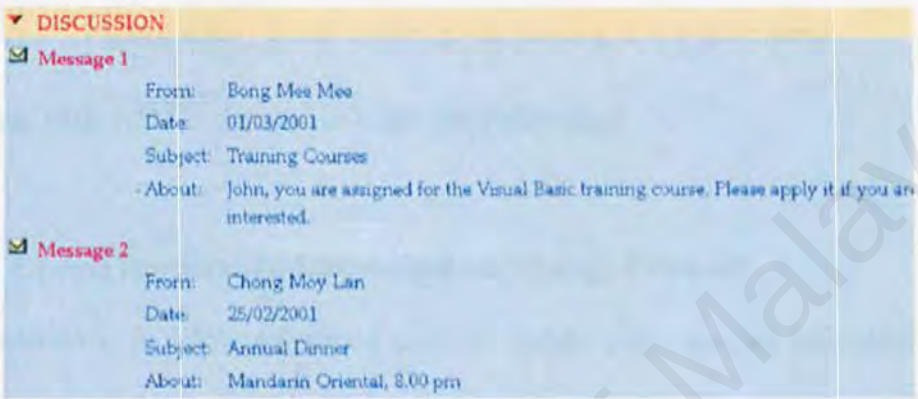

Figure 2.6.2 Discussions

4) To post a new discussion, users can click Post New Discussion at the bottom of the page. Then, a form for new discussion is showed as in Figure 2.6.3.

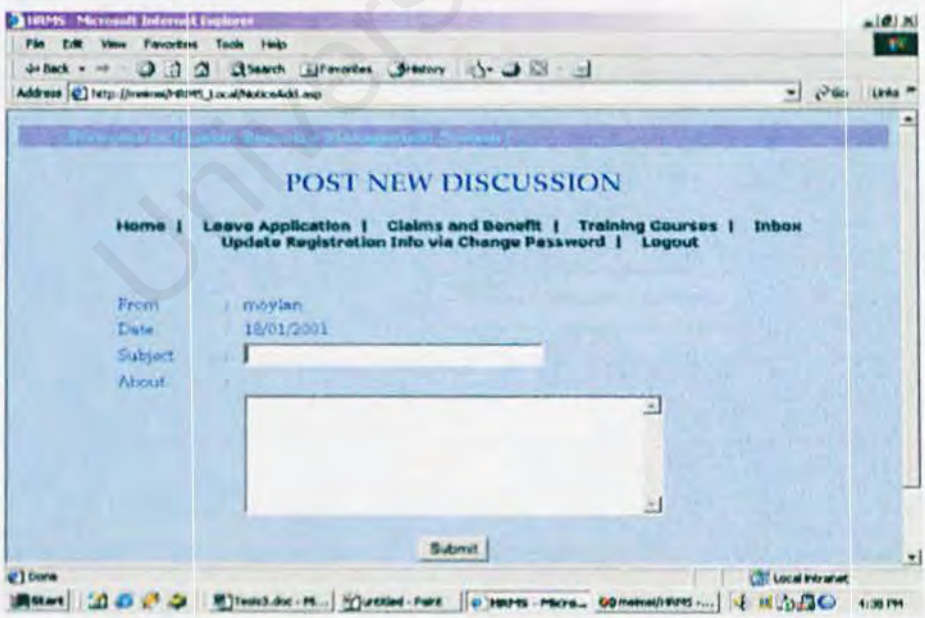

Figure 2.6.3 Post new discussion

6) Fill in the subject of the discussion and content of the discussion.

7) If the discussion does not contain a subject or the content, error messages will be displayed.

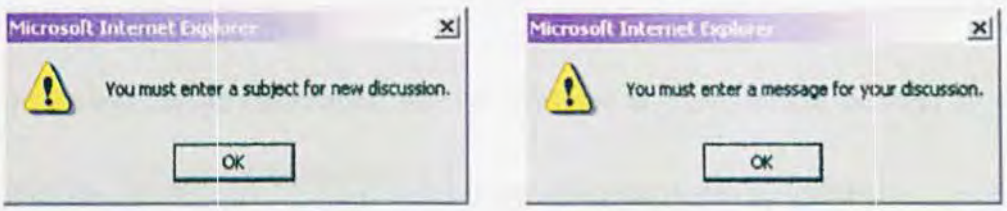

Figure 2.6.4 Error messages in posting a new discussion

8) Then, click Submit to add to it the discussion page.

2.7 Update Registration Information via Change Password

This module is to allow registered users to update their personal information and change their password if necessary.

1) To update, click **Update Registration Info via Change Password**. You will see the page as shown in Figure 2.7.1.

Figure 2.7.1 Update user registration via change password

2) Make changes to those information needed and verify your password.

3) If you want to maintain your old password, verify your password with the old one. But, if you wish to change it to a new password, change the current password to the new one and verify it.

4) Make sure that all information with "*" are filled in. Then, submit the form by clicking Submit Registration. Your information will be updated successfully as below.

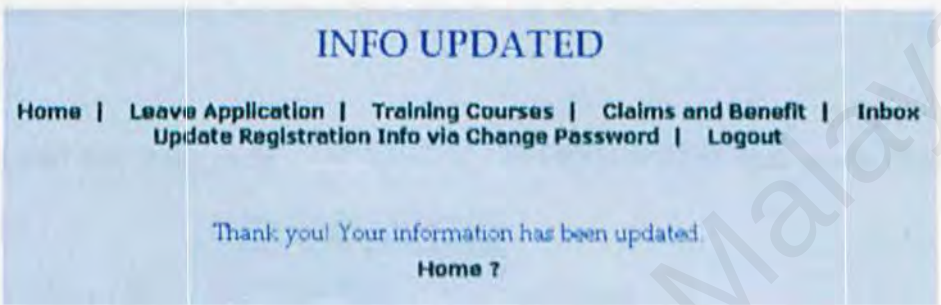

Figure 2.7.2 User information updated message

PART 3: ADMINISTRATOR MODULE

In an administrator module, there will be some differences with the non-administrator module. A few functions are added, which are hidden from the non-administrator's view.

3.1 Search employee information

Administrators can search for certain employee and the objective of searching is to view their information that comprises of personal and work information.

1) To start searching, click **Search Employee Information** that can be found at any page or click **Search Employee Information** at "Home."

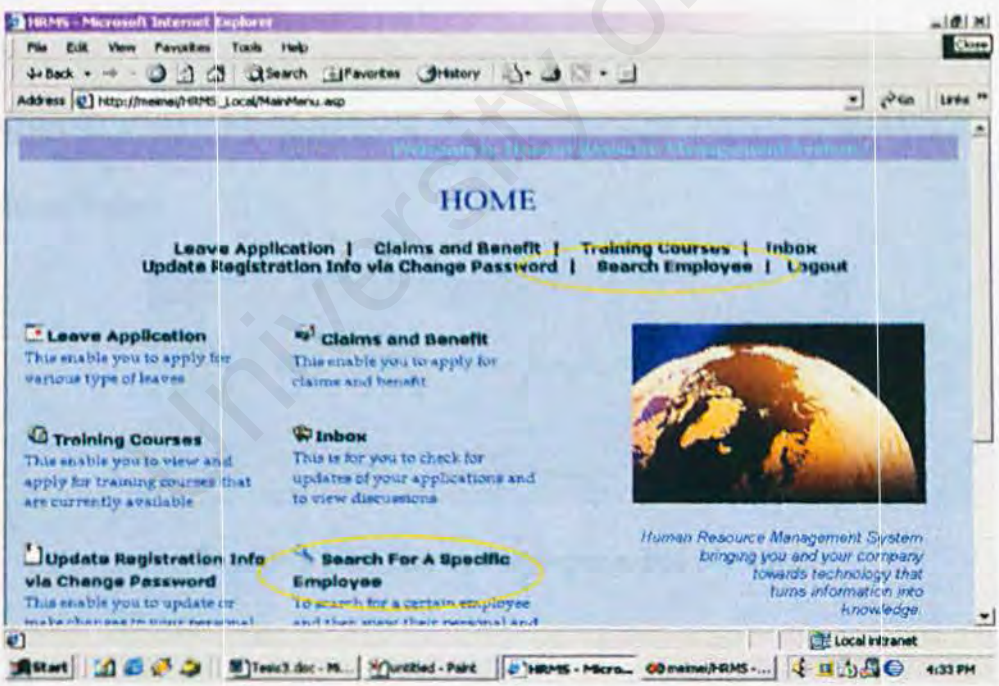

Figure 3.1.1 Links to search employee information

2) After clicking the link, search employee form will be shown as in Figure 3.1.2.

Figure 3.1.2 Search employee form

3) Now, users can start searching by entering at least one of the information required in the form. No information entered will result in error message to display as shown below:

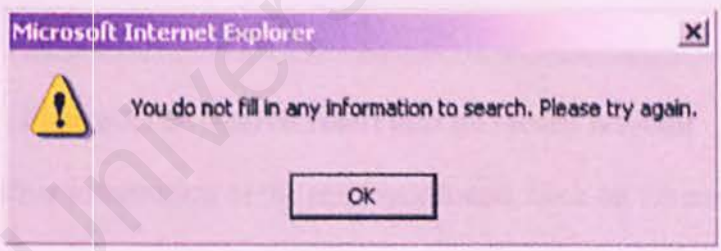

Figure 3.1.3 Search employee error message

4) Click Search Now to search.

5) The result of searching is either no record to be found or with records needed. Figure 3.1.4 shows the search result if a record is found.

Figure 3.1.4 Search result

6) If no record is found, a message is displayed to inform about the search result, as shown in Figure 3.1.5.

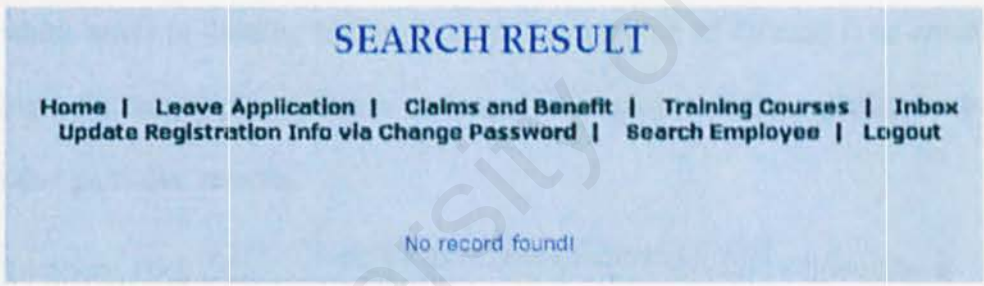

Figure 3.1.5 Search result that no record is found

7) To view further information of the employee found, click on the employee's name as shown below.

Figure 3.1.6 To view the employee's further information

8) Then, the employee's information is displayed as in Figure 3.1.7.

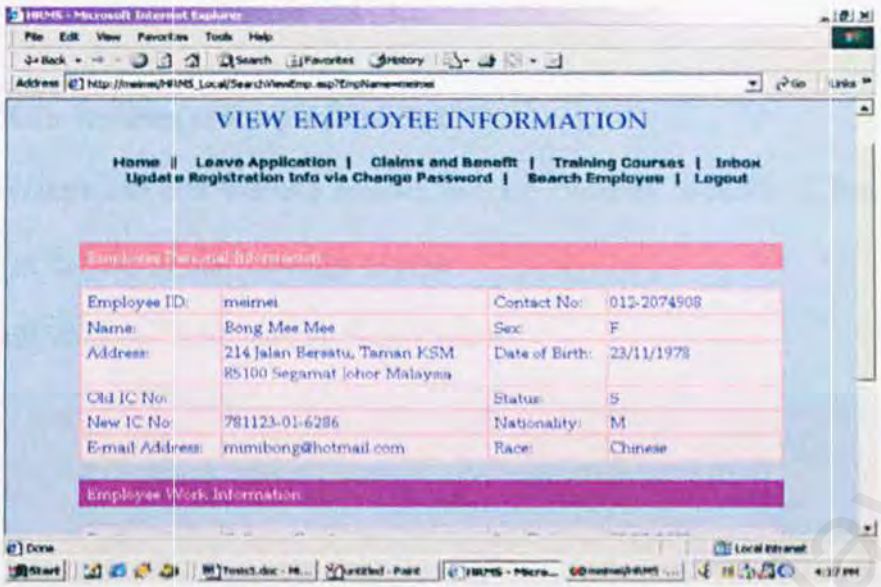

Figure 3.1.7 Search for specific employee from search result

8) Finally, administrators have the authority to delete the employee's information, which refers to deleting his/her account. The purpose of deleting is to cancel the login for that employee that is no longer an employee of the organization or for other particular reasons.

9) To delete, click Delete this employee's account? followed by a confirmation message as below:

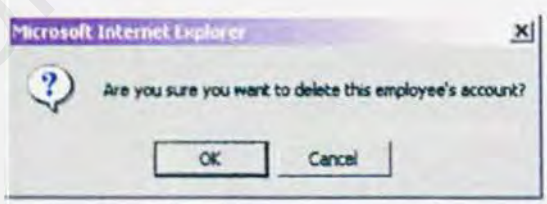

Figure 3.1.8 Delete employee confirmation message

10) Click OK will delete the employee's information successfully, whereas click Cancel will not cancel the deletion.

11) Click Back is to return to the previous screen, which is the search result.

3.2 Edit the training courses information

Administrators can edit training courses that are currently available if there is any changes or updates to that particular courses.

1) To edit, click on "Edit" link as circled below.

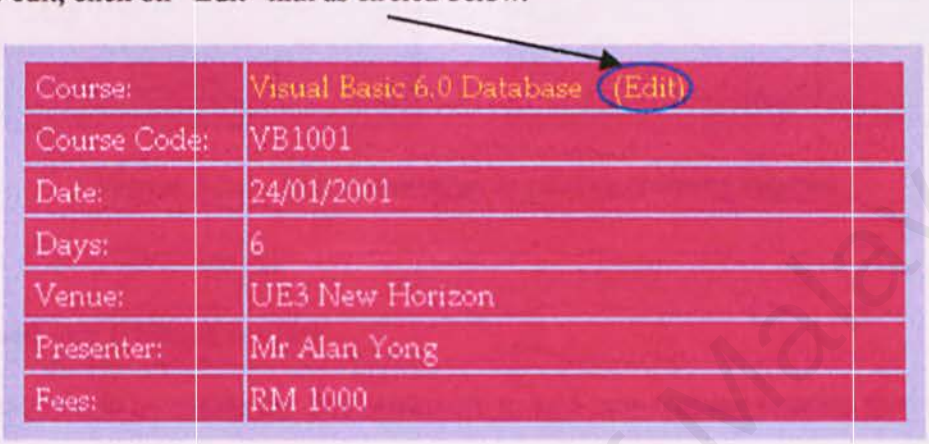

Figure 3.2.1 Link to edit training course

2) Then, make changes to any information that needs modification to the form as appeared below. (Figure 3.2.2)

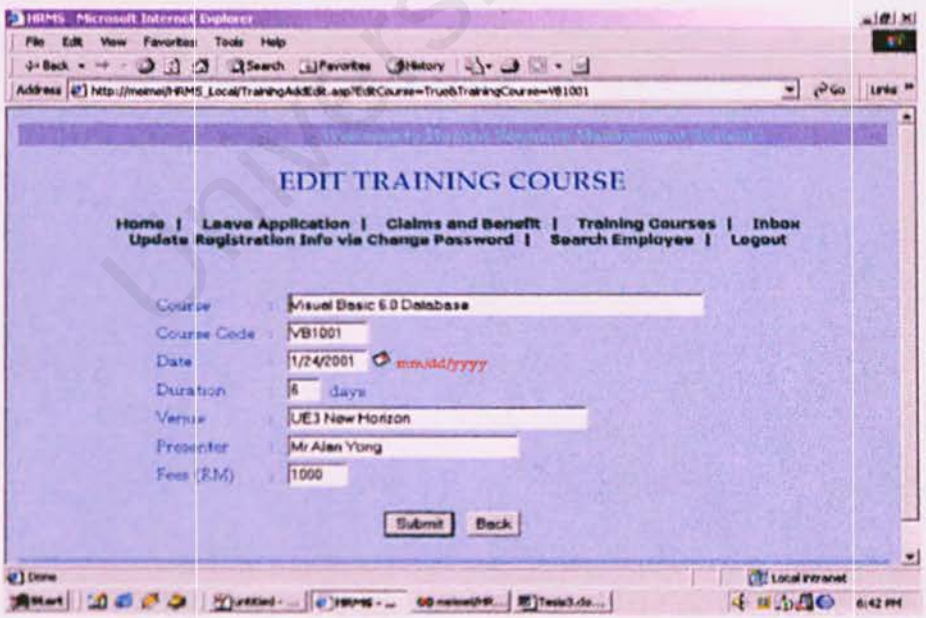

Figure 3.2.2 Edit training course form

3) Then, click Submit to update the changes made or click Back to return to the previous page without saving the changes to the course.

4) If the information are not filled in completely upon submitting, error messages will be displayed. Examples of error messages are:

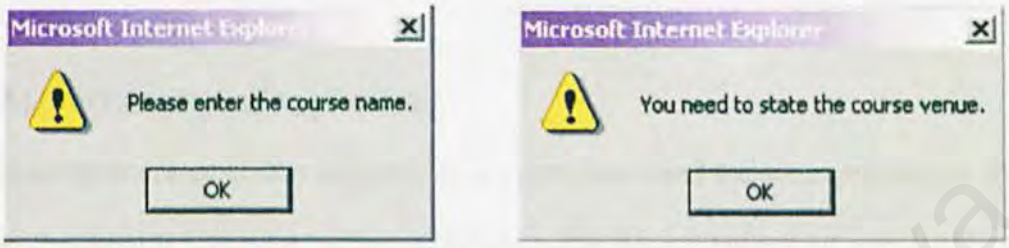

Figure 3.2.3 Error messages in editing training courses

3.3 Add new training courses

This module is to enable the administrators to add new training courses that will be available to the employees.

1) To add a new training course, click Add New Course at the bottom of "Training Courses" page. Then, the page below is shown. (Figure 3.3.1)

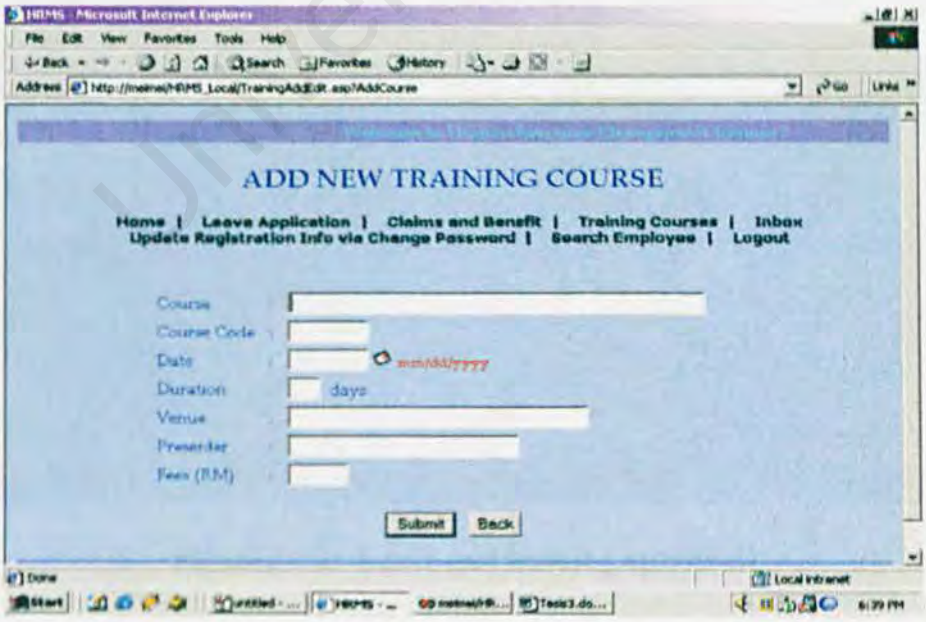

Figure 3.3.1 Add new training course

2) Fill in all the particulars required and click Submit to submit it or click Back to return to the previous page without saving the new training course.

3) If the particulars are not filled in completely, error messages will be prompted as in Figure 3.2.3.

3.4 Approve employees' applications

Administrators are given the authority to approve leave and training applications for the employees that are under his/her supervision. The steps to approve it are:

1) Go to "Inbox" to view the list of employees that have applied for any leaves or trainings in "Personal Items" section.

2) Click on "Approve" to approve for the applications. There is a warning mentioned that click on "Approve" will approve the applications immediately without prior confirmation message. Below is a part of "Inbox" where approvals are made.

Figure 3.4.1 Leave and training approvals

3) Administrators will be informed that a particular application is approved as in Figure 3.4.2 and Figure 3.4.3.

Figure 3.4.2 Leave approved

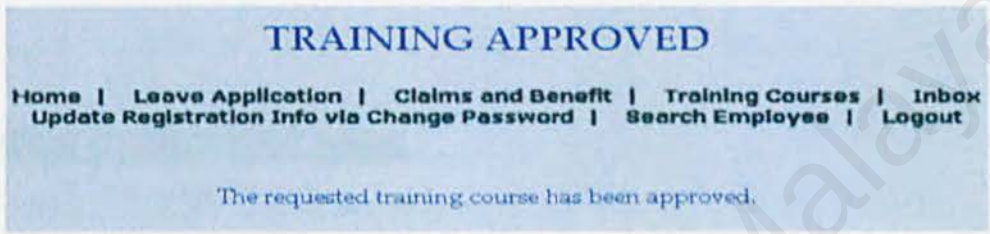

Figure 3.4.3 Training approved

4) After the leave or training is approved, the application status will change as shown below:

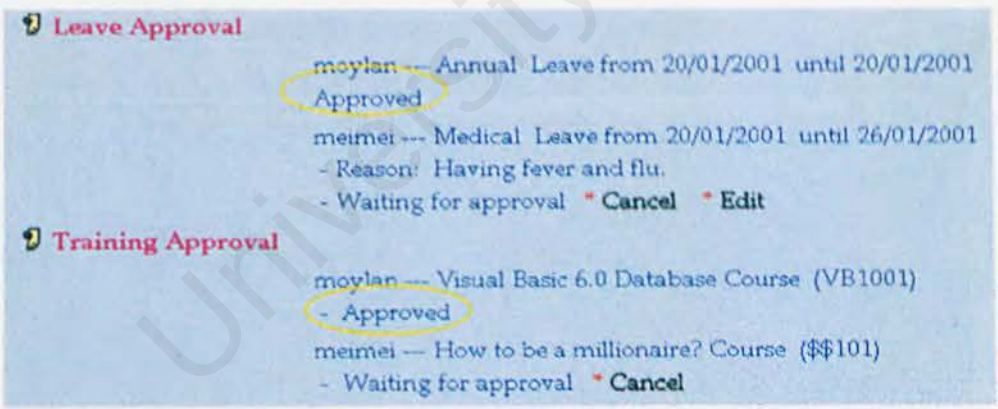

Figure 3.4.4 Feature of approved leave and training applications

3.5 Delete overdue items and old discussions

Overdue leave and training applications in database should be deleted, as well as old discussions. Administrators are allowed to do so when necessary. To delete,

1) Go to "Inbox".

2) Click * Delete overdue items to delete leave and training applications overdue.

3) Click * Delete old discussions to delete old discussions.

4) Again, "*" is to warn that all deletion are without prior confirmation message.

5) Deleted overdue items and old discussions will refresh the current page.

PART 4: QUITTING HRMS

4.1 Logout

To logout out, click **Logout** that will bring you to logout page as below and redirect to the login page automatically.

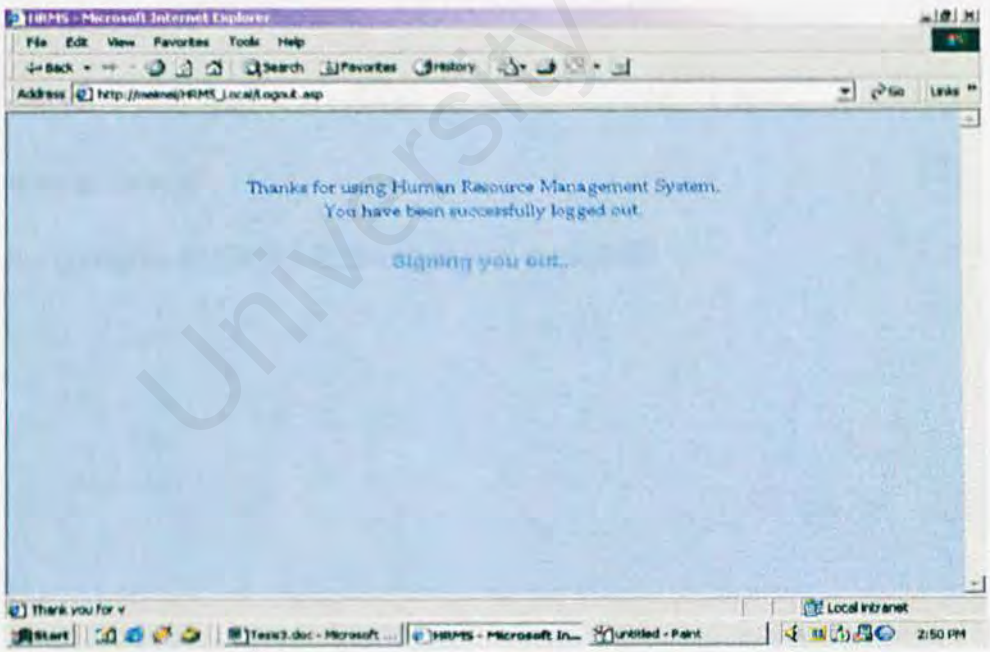

Figure 4.1.1 Logout page